From Recursive Depth to Dynamical Systems:

A Formalization of Eglash's Recursive-Depth Measure of Generativity

Wanhong Huang*

June 2026

(working draft)

Working draft • for reference only. Some arguments may not yet be fully rigorous; please read accordingly. Comments and ideas are warmly welcome at `huangwanhong@serendip.ngo` (Wanhong Huang).

Abstract

We give a rigorous formal-language and dynamical-systems foundation for Eglash's notion of *recursive depth*, a measure of generativity from the theory of generative justice [1]. First, we show that the recursive-depth equation is not merely analogous to, but *is*, a parsing score: it equals the root inside score of a weighted (probabilistic) context-sensitive grammar, with each of its symbols (d, r, P, E, W, M) receiving a coherent grammatical interpretation, and it admits a closed form as a sum over root-to-leaf descent paths. The environmental-response factor E forces the weighting to depend on derivation context, placing the parse at the (weighting-sense) context-sensitive level of the Chomsky hierarchy. Second, we give a mechanical, structure-preserving conversion of any such grammar into a multitype branching process, under which derivations and population evolution are equal in distribution; the spectral radius λ of the resulting fertility operator gives a dynamical criterion separating regeneration ($\lambda > 1$) from extraction ($\lambda < 1$). A consequence invisible to the static score is that *recursive depth is not regeneration*: an acyclic nesting has $\lambda = 0$, and genuine self-maintenance requires the loop network to close into a cycle of net gain exceeding one. The bridge places the analytic methods of both generative grammar and dynamical-systems theory at the service of generativity.

Keywords: generative justice; recursive depth; weighted context-sensitive grammar; semiring parsing; multitype branching process; spectral radius; regeneration.

*`huangwanhong@serendip.ngo`

Recap (the formalism at a glance). Let G be a (weighted) context-sensitive grammar with parent–child edge weight $w_{i\to j} = W_j\, d_j\, P_{j-1}\, E_j$.

Claim 1 (parsing). Eglash's recursive depth R is realized as the root inside score of G, which admits the root-to-leaf path-sum closed form

$$R = \beta(S) = \sum_{\pi:\, S\to\text{leaf}} \prod_{(i\to j)\in\pi} w_{i\to j}.$$

The dependence of the weight on context through E pushes the parse to the (weighted) *context-sensitive* level.

Claim 2 (dynamics). Reading $w_{i\to j}$ as an *expected number of offspring* turns G into a multitype branching process with fertility matrix $\mathsf{M} = (w_{i\to j})$. The same R is then its expected total progeny,

$$R = \mathbf{e}_S (I - \mathsf{M})^{-1}\mathbf{1} \quad (\text{finite iff } \lambda < 1), \qquad \lambda := \rho(\mathsf{M}),$$

and the spectral radius λ is the asymptotic growth rate:

$$\text{regime} = \begin{cases} \text{self-sustaining} & \lambda > 1, \\ \text{critical} & \lambda = 1, \\ \text{extractive} & \lambda < 1. \end{cases}$$

Depth does not entail regeneration: large $R \not\Rightarrow \lambda > 1$; an acyclic type graph has $\lambda = 0$ regardless of R, so genuine regeneration requires the loop network to *close into a cycle* with net gain > 1.

Notation (plain-language gloss, for readers new to formal-language notation).

- $A \to \alpha$: a *production*, "A may be rewritten as α."
- $\oplus, \otimes$: the semiring's "add" and "multiply" (here ordinary $+, \times$); $\bigoplus, \bigotimes$ are their repeated forms (like $\sum, \prod$).
- $|$: "given" (conditional on); $\varnothing$: the empty context; $\#\{\cdot\}$: "the number of."
- $\rho(\mathsf{M})$: the *spectral radius* (largest eigenvalue magnitude), here the growth rate λ; $\mathrm{spec}(\cdot)$: the set of eigenvalues.
- $\gtrless 1$: shorthand for the three cases "> 1, $= 1$, or < 1."
- $(I - \mathsf{M})^{-1} = \sum_{t\geq 0} \mathsf{M}^t$: the "total over all generations."
- $\mathbf{e}_S$: starts the count at the root S; $\cong$: "corresponds to."
- "$\hookrightarrow$" (in the mapping tables): a sub-factor of the row above.

Contents

1 Claim 1: Recursive Depth as PCSG Parsing **4**

1.1 Claim 1: the reduction proposition 4

1.2 Definitions of grammar and parsing 4

1.2.1 Intuitive illustration: what the inside score does 6

1.3 The reduction theorem and its proof 8

1.4 Conceptual mapping 11

1.5 Discussion and analysis 11

2 Claim 2: Mechanical Conversion of a PCSG into a Dynamical System 15
2.1 Claim 2: the conversion proposition 15
2.2 Definition of the target class of dynamical systems 15
2.3 Construction of the mechanical conversion Φ 16
2.4 Distributional-consistency theorem 18
2.5 Spectral-criterion theorem: regeneration is spectrum, depth is not regeneration . 19
2.6 Continuous-time limit and basins of attraction 20
2.7 Conceptual mapping 21
2.8 Discussion and analysis 21

1 Claim 1: Recursive Depth as PCSG Parsing

1.1 Claim 1: the reduction proposition

In Appendix A of the chapter "Foundational Principles of Generative Justice," Eglash [1] gives a computable measure of "generativity," called the *recursive depth* R, defined by

$$R \;=\; \sum_{i=1}^{N} d_i \, r_i \, P_{i-1} \, E_i, \tag{1.1}$$

$$r_i \;=\; \sum_{j=1}^{M_i} W_j \, d_j \, r_j \, P_{j-1} \, E_j. \tag{1.2}$$

In the original text this measure is presented as a score for a "nested-loop structure." The goal of this section is to lift that presentation into a *formally provable* proposition.

Claim 1 (Grammatical-parsing reduction of recursive depth)**.** Eglash's recursive-depth equations (1.1)–(1.2) can be rigorously reduced to a *parsing-score* computation over a *probabilistic (weighted) context-sensitive grammar* (hereafter **PCSG**). That is: there exist a weighted context-sensitive grammar G, a semiring $\mathbb{K}$, and a context-dependent weight assignment w, such that the root inside score $\beta(S)$ of G from the start symbol is identically equal to R; and each symbol of R (d, r, P, E, W, M) acquires, under this reduction, a grammatically and semantically coherent counterpart. Moreover, context-sensitivity *in the weighting sense* is *necessary*: the environmental response efficacy E corresponds exactly to the dependence of the weight on the derivation context, so that once E is taken seriously, the score cannot be reproduced by *any* context-independent weight assignment (i.e. a weighted context-free grammar, PCFG), and is necessarily a context-*dependent* parse (hereafter PCSG). One point must be clarified in advance to avoid a common confusion: by "context-sensitive" we mean the dependence of the *weight function* on the derivation context (a property of the *score*), not the context-sensitivity of the Chomsky hierarchy regarding *string-language generative power* – the latter is not claimed by this reduction; see Remark 1. In short, E is precisely the factor that pushes this parse beyond PCFG.

The remainder of this section gives, in order: the required definitions of grammar and parsing (§1.2); the reduction theorem and its proof (§1.3); the symbol-by-symbol conceptual mapping (§1.4); and the structural analysis exposed by the reduction (§1.5).

1.2 Definitions of grammar and parsing

Definition 1 (Semiring)**.** A *semiring* $\mathbb{K} = (K, \oplus, \otimes, \mathbf{0}, \mathbf{1})$ consists of a set K and two binary operations $\oplus$ (sum) and $\otimes$ (combination) such that $(K, \oplus, \mathbf{0})$ is a commutative monoid, $(K, \otimes, \mathbf{1})$ is a monoid, $\otimes$ distributes over $\oplus$ on both sides, and $\mathbf{0}$ is absorbing for $\otimes$. The *probability semiring* is $(\mathbb{R}_{\geq 0}, +, \times, 0, 1)$. Strictly speaking, the scores on this semiring are genuine *probabilities* only when the production weights additionally satisfy a normalization constraint; the R of this reduction does *not* satisfy it (see §1.5, item (v)). We therefore use "probability semiring" only for its $(+, \times)$ algebraic form; it may equally be called the *sum–product* or inside semiring.

Scores accumulate $(\oplus)$ and combine $(\otimes)$ over a semiring; different semirings give different parsing semantics (the probability semiring gives the total inside score; the tropical semiring

$(\mathbb{R} \cup \{\infty\}, \min, +)$ gives the Viterbi optimal parse). Since (1.1)–(1.2) literally use $\sum$ and $\times$, we fix $\mathbb{K} = (\mathbb{R}_{\geq 0}, +, \times, 0, 1)$ below.

Definition 2 (Weighted context-sensitive grammar, PCSG)**.** A *weighted context-sensitive grammar* is a six-tuple

$$G = (V,\ \Sigma,\ \mathcal{R},\ S,\ \mathbb{K},\ w), \tag{1.3}$$

where V is the set of nonterminals, Σ the set of terminals ($V \cap \Sigma = \varnothing$), $S \in V$ the start symbol, $\mathbb{K}$ a semiring, and every rule in the production set $\mathcal{R}$ has the context-sensitive form

$$\gamma\, A\, \delta \ \longrightarrow\ \gamma\, (B_1 B_2 \cdots B_M)\, \delta, \qquad A \in V,\ \gamma, \delta \in (V \cup \Sigma)^*,\ M \geq 0, \tag{1.4}$$

i.e. the nonterminal A may expand into the substring $B_1 \cdots B_M$ only within the environment formed by the left context γ and the right context δ. The weight function

$$w :\ \mathcal{R} \times \mathrm{Ctx} \ \longrightarrow\ K, \qquad (A \to B_1 \cdots B_M) \mapsto w(A \to B_1 \cdots B_M \mid \gamma, \delta) \tag{1.5}$$

depends explicitly on the left and right contexts $(\gamma, \delta) \in \mathrm{Ctx}$. If and only if this weight takes the same value for all (γ, δ) (i.e. w is context-independent) does G degenerate to a weighted context-free grammar (PCFG).

Remark 1 (Two senses of "context-sensitive" – please do not conflate them)**.** "Context-sensitive" has two non-coinciding meanings in the literature; we claim only one.

(a) **Generative-power sense (Chomsky hierarchy):** whether the grammar can generate string languages such as $a^n b^n c^n$ that a *context-free grammar cannot.* This concerns the *rewriting power of productions.*

(b) **Weighting sense (what we claim):** whether, over a *structural skeleton* (here a context-free branching structure), the *weight function* depends on the derivation context. This concerns *scoring*, not string generation.

This reduction constrains rewriting to a branching (tree-shaped, context-free) skeleton (Definition 3); hence the underlying *string language remains context-free* – we do *not* claim sense (a). But the weight, through E, depends substantively on context, so as a *scoring function* R lies outside PCFG: no context-independent weight assignment can reproduce it. For a reduction that reads the equation as a *parsing score*, sense (b) is exactly the relevant "context-sensitivity." A reader expecting sense (a) will not find it here; this very distinction is one of the things the reduction clarifies.

Definition 3 (Branching derivations, nested structure, and descent paths)**.** Associate to each loop A a set of M_A *unary* productions

$$A \longrightarrow B_j \qquad (j = 1, \ldots, M_A), \tag{1.6}$$

i.e. "loop A may descend one step to its j-th subloop B_j" (this is the $M = 1$ special case of (1.4): the rewriting itself is context-free and unary, the context-sensitivity entering only through the weight $w(A \to B_j \mid \gamma, \delta)$). The *nested structure* τ of the system is the tree rooted at S obtained by hanging, under each loop, *all* of its subloops $B_1, \ldots, B_{M_A}$; it depicts the grammar's *reachable structure* (a purely *disjunctive*/OR tree), not any single derivation. A single *derivation* descends

from the root, at each step choosing *one* of the current loop's subloops, down to a leaf; thus a derivation is exactly a *root-to-leaf path* in τ. The contexts (γ, δ) do not alter the parent–child relations of τ, but only *modulate* the weight carried by each edge. Write $\mathcal{N}(\tau)$ for the node set of τ, with leaves being the loops without subloops (labels in Σ).

Remark 2 (Why "disjunctive/OR" rather than "conjunctive/AND")**.** The unarization here is not arbitrary: it is *forced* by the form of Eglash's inner equation (1.2). That equation *adds* the subloop scores r_j ($\sum_j$), it does not *multiply* them. In semiring parsing, *summing* over child terms means they are *alternative* (disjunctive) branches, corresponding to "the loop descends into one of its subloops"; only *multiplying* over child terms would mean they *coexist conjunctively* (a single production $A \to B_1 \cdots B_M$). In other words, once the equation is written with $\sum$, it has already selected the OR reading and fixed R as "the sum over all descent paths" (Corollary 1). If Eglash's intent were instead "all subloops coexist, their scores multiplied," the equation should use $\prod$; this is a substantive choice between the equation and the intuition, deferred to §1.5, item (ii), to be settled by Eglash. Under either reading, context-sensitivity *enters only the weights* and does not break the topological correspondence "nested loops $\cong$ tree" – which is precisely the pivot on which the reduction holds while genuinely residing at the (weighting-sense) context-sensitive level.

1.2.1 Intuitive illustration: what the inside score does

Before the formal definitions, a minimal example conveys the whole idea of the inside score. The terminology sounds complicated, but the algorithm is one sentence:

> **The score of each loop = take each of its subloops' scores, multiply by "the weight of the edge connecting them," and sum. A bottom leaf loop has no subloops, and its score is defined to be 1. Starting from the very bottom, multiply and add layer by layer up to the outermost root; the root's score is R.**

Three terms in plain words (for readers unfamiliar with formal languages). For convenience, one sentence on each of the three recurring terms. **Parsing**: given a structure, compute its "score" *bottom-up* by rules – which is exactly what Eglash's two recursions do. **Semiring**: a *minimal* algebra requiring only "one addition $\oplus$ and one multiplication $\otimes$," enough to make "multiply along a path, add across paths" well-defined; we take ordinary arithmetic $(+, \times)$, so $\oplus$ is addition and $\otimes$ is multiplication. Choosing a different semiring merely changes the "how to add, how to multiply" convention (e.g. replacing "add" by "take the smaller" turns the "total" into the "single best" – see §1.5, item (iv)). **Inside score**: the score accumulated by each node "looking *inward*, toward its subtree"; a leaf scores 1, and a parent multiplies each child's score by "the weight of the connecting edge" and sums. The name comes from its using only the information "*inside* the node (the subtree)." The minimal example below says all three at once.

Imagine a worker cooperative (root S) containing two subloops: a "labor–wage" loop A and an "ecological exchange with the community" loop B; A itself contains a leaf loop a (learning some skill). Its nested structure is shown in Figure 1.

Now compute the inside score. To see the *algorithm itself*, assign an arbitrary example weight to each edge (the edge weight $= W\,d\,P\,E$, here just sample values): the root-to-A edge weight

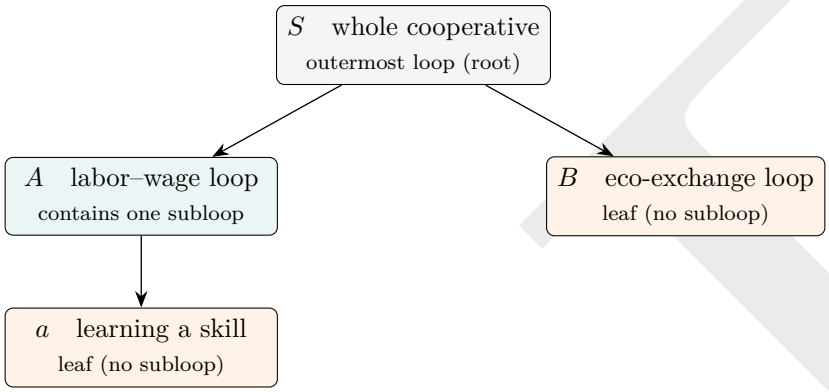


Figure 1: Nested-loop structure of the cooperative system: the root S has children A and B; A has the leaf a. Both B and a are leaves.

is 2, the root-to-B edge weight is 5, and the A-to-a edge weight is 8. The computation proceeds bottom-up; see Figure 2.

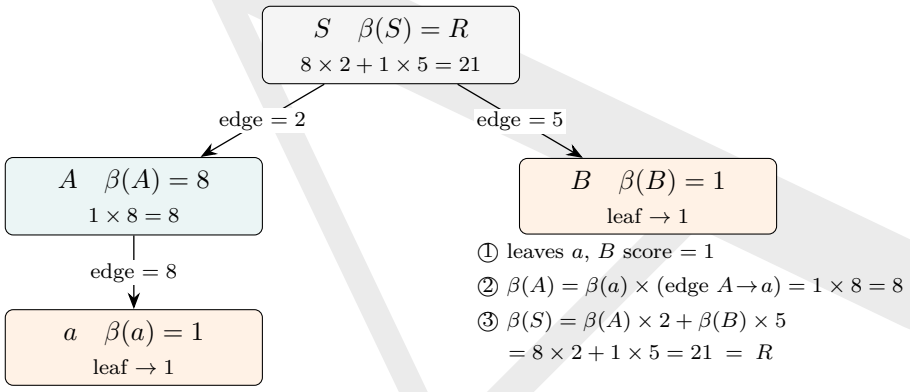


Figure 2: Bottom-up computation of the inside score. Leaves score 1; each parent's score = the sum over children of (child score × edge weight); rolling up to the root gives R.

This corresponds exactly to Eglash's two equations: the inner equation (1.2) is "multiply each subloop's score r_j by the edge weight $W_j d_j P_{j-1} E_j$ and sum," and the outer equation (1.1) does the same thing at the root. Hence

$$\text{“inside score”} = r_i, \qquad \text{“root inside score”} = R. \tag{1.7}$$

The reduction is nothing but the observation that this "multiply and add up from the subloops" algorithm is precisely the standard algorithm used for decades to parse a grammar tree in formal language theory (the inside algorithm). We now formalize it.

Definition 4 (Context-sensitive inside score)**.** For a node v in the derivation-tree skeleton τ with label A, production $A \to B_1 \cdots B_M$, and context (γ, δ), define its *inside score* $\beta(v \mid \gamma, \delta) \in K$ by

$$\beta(v \mid \gamma, \delta) \;=\; \bigoplus_{A \to B_1 \cdots B_M} \left[w(A \to B_1 \cdots B_M \mid \gamma, \delta) \;\otimes\; \bigotimes_{k=1}^{M} \beta\big(\,\mathrm{child}_k(v) \mid \gamma_k, \delta_k\big) \right], \tag{1.8}$$

where (γ_k, δ_k) is the local context inherited by the k-th child in the derivation; a leaf takes $\beta(\text{leaf}) = \mathbf{1}$. The score of the whole derivation tree is the root score $\beta(S \mid \varnothing)$.

Remark 3 (Two technical notes: context propagation; and how (1.8) reduces to a sum over subloops here)**.** *(i) Context propagation.* For (1.8) to be well-defined, one must additionally specify a *context-propagation function* $(\gamma, \delta) \mapsto (\gamma_k, \delta_k)$ prescribing which local context each child inherits from its parent. We leave this function unspecified (different coupling structures correspond to different propagation rules), requiring only that it be well-defined; Theorem 1 holds for any fixed propagation rule, because the value of $E_j(\gamma, \delta)$ does not affect the algebra of the reduction.
(ii) From "unary rules" back to the sum. Here each loop corresponds to a set of *unary* rules $A \to B_j$ (Definition 3). Hence in (1.8) the product over children $\bigotimes_{k=1}^{M}$ is *trivial* since $M = 1$ (each rule hangs a single child), while the sum over rules $\bigoplus_{A \to B_j}$ realizes exactly "sum over the M_A subloops of A." This is precisely the $\sum_j$ of Eglash's inner equation (1.2), i.e. the edge-weighted inside recursion (1.11) below. In one sentence: in the "purely disjunctive branching" special case, the standard inside algorithm automatically degenerates to a weighted sum over subloops.

Formula (1.8) is semiring parsing in Goodman's sense [2]; over the probability semiring it becomes

$$\beta(v_i \mid \gamma, \delta) = \sum_{\text{rule}} w(\cdot \mid \gamma, \delta) \prod_{j} \beta(v_j \mid \gamma_j, \delta_j). \tag{1.9}$$

1.3 The reduction theorem and its proof

We first give the key observation that makes the reduction work: in (1.2), the subloop score r_j is multiplied by a coefficient $W_j d_j P_{j-1} E_j$ that *depends term-by-term on* j; whereas in the standard inside recursion (1.9) the parent weight $w(\cdot)$ is pulled outside the sum and shared by all child terms. The two are equal if and only if the parent loop's weight is "pushed down" onto each parent–child edge, i.e. one adopts the *edge-weighting* form. The following lemma formalizes this.

Lemma 1 (Edge-weighting rewrite)**.** *Over the probability semiring, define for each parent–child edge $i \to j$ the edge weight*

$$w_{i \to j}\,(\gamma, \delta) \;:=\; W_j\, d_j\, P_{j-1}\, E_j(\gamma, \delta), \tag{1.10}$$

where the dependence of E_j on the context (γ, δ) is made explicit. Then the inner recursion (1.2) *is equivalent to the edge-weighted inside recursion*

$$\beta(v_i \mid \gamma, \delta) \;=\; \sum_{j=1}^{M_i} w_{i \to j}(\gamma, \delta) \;\cdot\; \beta(v_j \mid \gamma_j, \delta_j), \qquad \beta(v_i) \equiv r_i. \tag{1.11}$$

Proof. Substituting (1.10) into the right-hand side of (1.11):

$$\sum_{j=1}^{M_i} w_{i \to j}(\gamma, \delta)\, \beta(v_j) = \sum_{j=1}^{M_i} (W_j d_j P_{j-1} E_j)\, \beta(v_j). \tag{1.12}$$

Setting $\beta(v_j) = r_j$ (induction hypothesis: a child's inside score is its recursive depth), the right-hand side is $\sum_{j=1}^{M_i} W_j d_j r_j P_{j-1} E_j$, which agrees term-by-term with the right-hand side of (1.2). The base case is a leaf loop $r_{\text{leaf}} = \mathbf{1} = 1$, consistent with "the internal recursive depth is the multiplicative unit when there is no subloop." Hence $\beta(v_i) = r_i$ for every node. □

Theorem 1 (Reduction theorem for Claim 1)**.** *Suppose the system's nested-loop structure corresponds to a derivation-tree skeleton τ of a weighted context-sensitive grammar G (Definition 2) under a branching derivation (Definition 3), where: each loop corresponds to a node of τ, the outermost loop to the root S, and the leaf loops to terminals. Take the probability semiring, and set the edge weights to*

$$w_{S\to i}(\varnothing) := d_i\, P_{i-1}\, E_i, \qquad (P_0 = 1), \tag{1.13}$$

$$w_{i\to j}(\gamma,\delta) := W_j\, d_j\, P_{j-1}\, E_j(\gamma,\delta). \tag{1.14}$$

Then the root inside score reproduces the recursive depth exactly:

$$\boxed{\beta(S \mid \varnothing) \;=\; R} \tag{1.15}$$

where R is given by (1.1)–(1.2)*.*

Proof. Apply the edge-weighted inside recursion (1.11) at the root S and substitute the root edge weight (1.13):

$$\beta(S \mid \varnothing) = \sum_{i=1}^{N} w_{S\to i}(\varnothing)\, \beta(v_i) = \sum_{i=1}^{N} d_i\, P_{i-1}\, E_i\, \beta(v_i). \tag{1.16}$$

By Lemma 1, $\beta(v_i) = r_i$. Substituting gives

$$\beta(S \mid \varnothing) = \sum_{i=1}^{N} d_i\, P_{i-1}\, E_i\, r_i = \sum_{i=1}^{N} d_i\, r_i\, P_{i-1}\, E_i = R, \tag{1.17}$$

i.e. (1.1). Note that the root edge carries no W factor, consistent with the absence of W in the outer equation (1.1); the boundary condition $P_0 = 1$ corresponds to "no parent loop" at the root. □

Corollary 1 (Path-sum closed form for R)**.** *Under the edge weights of Theorem 1, the recursive depth equals the sum over* all root-to-leaf paths*, each path's score being the product of the edge weights along it:*

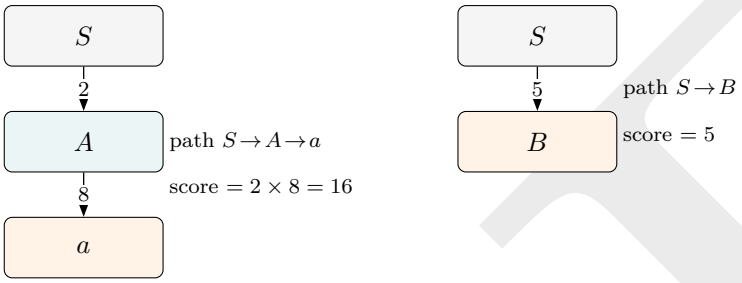
$$R = \beta(S \mid \varnothing) = \sum_{\pi:\, S\to \mathit{leaf}} \prod_{(i\to j)\in\pi} w_{i\to j}. \tag{1.18}$$

Proof. Unfold the edge-weighted inside recursion (1.11) repeatedly down to the leaves (where $\beta = \mathbf{1}$); by distributivity, the nested structure "first multiply by the edge weight, then sum over children" expands precisely to "sum over all root-to-leaf path-products." □

This gives R its plainest reading: **for each regenerative path in the system that "nests from the outermost layer all the way to the bottom," record a score equal to the product of its couplings along the way, then add up the scores of all paths.** In Figure 2 there are only two paths – $S \to A \to a$ (score $2 \times 8 = 16$) and $S \to B$ (score 5) – and their sum 21 is R. Figure 3 draws the two paths separately.

Remark 4 (Bridge to Claim 2: the path-sum is the expected total progeny, clarifying $\sum$ vs. $\prod$)**.** The path-sum closed form (1.18) is not merely a rewriting of R; it is the very entry point to Claim 2. Reading the edge weight $w_{i\to j}$ as "the expected number of type-j subloops born from a type-i loop,"

$$R = \sum_{\pi} \prod_{(i\to j)\in\pi} w_{i\to j} \;=\; \mathbf{e}_S \Big(\sum_{t\ge 0} \mathsf{M}_{\mathrm{wt}}^{t} \Big) \mathbf{1} \tag{1.19}$$



$R = 16 + 5 = 21$ (sum of the two paths' scores)

Figure 3: The path-sum reading of R (Corollary 1). The nested structure of Figure 1 is split into two independent root-to-leaf descent paths: record each path's product of edge weights, then sum over the paths to obtain R. "Summing" rather than "multiplying" is exactly the (disjunctive/OR) semantics Eglash selected by writing $\sum$.

is exactly the **expected total progeny** of a *multitype branching process* (where $\mathsf{M}_{\mathrm{wt}} = (w_{i \to j})$ is the expectation/fertility matrix). This observation at once settles the $\sum/\prod$ question left open in §1.5, item (ii): because expectation is *linear*, "summing over subloops" and "subloops coexisting" are not in conflict – the subloops are indeed simultaneously present (matching Eglash's intuition that "a parent loop contains subloops"), and we *still* sum their expected contributions. Hence Eglash's $\sum$ is *correct*, provided R means "expected regenerative throughput"; $\prod$ would answer a different question (the likelihood that one entire joint structure is realized). The earlier "ask Eglash to choose between OR and AND" thus tightens into a *recommendation*: $\sum$ is "expected regenerative output," equal in value to the path-sum, and acquires in Claim 2 its asymptotic growth rate $\lambda = \rho(\mathsf{M}_{\mathrm{wt}})$. **In particular, Claim 2 will prove that depth is not regeneration – a deep but acyclic nesting has $\lambda = 0$ (eventually dies out); genuine regeneration requires the loop network to close into a cycle.** This lifts the present section's worry about the superlinear reward of d into a decidable dynamical criterion.

Remark 5 (Honest annotation of two textual inconsistencies)**.** The reduction exposes two points in the original worth noting; neither affects the validity of Theorem 1, but both should be made explicit:

(i) **The counting origin of d is not uniform.** In the outer equation (1.1), d_i counts "2 for the parent's children, 3 for grandchildren"; in the inner equation (1.2), d_j is a local depth "starting from 1 relative to the parent loop i." In the reduction this shows up as the root edge and the inner edges using different depth counts; the theorem still holds, but d is not a uniformly defined global quantity.

(ii) **d as an independent multiplicative factor is a departure from the standard inside recursion.** The standard semiring-parsing inside recursion (1.9) has no independent "depth" factor – a node's depth relative to the root is already determined uniquely by its position in the tree, so multiplying by d double-counts depth information. Precisely: $R = \beta(S)$ if and only if d is absorbed into the edge weight (1.10); if one insists that d carry an independent "depth" meaning, then R equals "the standard weighted parse" tensored with a "depth-rescaling kernel." This is the entry point for the analysis below (§1.5) but is irrelevant to the reduction itself.

(iii) **The leaf base case must be reinterpreted from "empty sum" to "multiplicative unit."** For a leaf with no subloops ($M_i = 0$), the inner equation (1.2) literally gives an *empty sum* $= \mathbf{0} = 0$; executed literally, any leaf's $r = 0$ would zero out the whole tree's score along the multiplications, collapsing R to 0. The reduction therefore adopts the standard inside convention, taking the leaf base case to be the *multiplicative unit* $\beta(\text{leaf}) = \mathbf{1} = 1$ (cf. B, a both taken as 1 in Figure 2). This reinterpretation does not affect Theorem 1, but it is a supplementary stipulation of a base case *left unstated* in the original and should be made explicit – otherwise the equation cannot be started as written.

Corollary 2 (Necessity of the context-sensitive level)**.** *If E_j substantively encodes "the loop's response to its environment," and that environment includes parts of the derivation tree outside the node (other loops at the same level, or the coupling between the system and the outside), then the edge weight* (1.14) *depends substantively on the context (γ, δ), whence, by the last sentence of Definition 2, G cannot degenerate to a PCFG. Therefore the parsing reduction of R* necessarily *resides at the* weighting-sense *context-sensitive level (PCSG). To emphasize again (Remark 1): this is an assertion about the dependence of the* weights *on context, not about string-language generative power; we claim "no context-independent weight assignment can reproduce R," not "G can generate string languages a context-free grammar cannot."*

Proof. By Definition 2, G degenerates to a PCFG iff w is constant in (γ, δ). If $E_j(\gamma, \delta)$ varies with context (i.e. the "response to the environment" differs across environments, which is exactly what the semantics of E requires), then $w_{i \to j}$ is non-constant, so G does not degenerate to a PCFG. The parse over $\mathbb{K}$ is therefore a genuinely context-sensitive parse. □

1.4 Conceptual mapping

Theorem 1 not only reproduces R numerically; it also requires each original symbol to acquire a grammatically coherent counterpart. Table 1 checks them one by one.

The overall picture of the mapping is clear: **the parent–child structure, P, W, M land cleanly in the (context-free part of the) weighted grammar's edge weights and productions; E cannot land in the context-free part and lands precisely as the weight's context-dependence, thereby pushing the whole parse into the context-sensitive level; while d has no counterpart and is redundant or a departure.** In other words, the reduction not only "can be done," but also *diagnoses* which parts of the original equation are grammatically necessary (P, W, M and the tree structure), which are forced by environmental coupling ($E \Rightarrow$ context-sensitive), and which are superfluous (d).

1.5 Discussion and analysis

(i) The division of labor between P and E: intra-tree vs. extra-tree coupling

The reduction reveals a distinction that is inconspicuous in the original equation but structurally fundamental. P_{j-1} (response to the parent loop) describes a parent–child edge *within the derivation tree*; it lies entirely within the context-free framework – parent–child is a tree edge, independent of the rest of the tree. By contrast, E_j (response to the environment) describes the node's coupling with the *environment outside the tree*: the parsing score of a context-free grammar *cannot in principle depend on anything outside the derivation tree.* Hence the only way to formalize E seriously is to let the weight depend on context (Corollary 2).

This yields a clean, provable, semantically grounded demarcation:

$$\underbrace{P}_{\text{inward, context-free}} \quad \text{vs.} \quad \underbrace{E}_{\text{outward, forces context-sensitivity}} . \tag{1.20}$$

The demarcation is not imposed by us: it is intrinsic to the respective semantics of P and E. Eglash juxtaposes the two as symmetric "response efficacy" factors, but the reduction shows that at the level of formal languages they are *not* symmetric – one is context-free, the other drags the whole system into context-sensitivity.

(ii) Sum or product: are subloops "alternative" or "coexisting"?

The reduction forces a semantic choice the original equation never makes explicit. The inner equation (1.2) *adds* the subloop scores ($\sum_j$). In semiring parsing, **summing over child terms = the terms are *alternative* (disjunctive/OR)**, corresponding to "the loop descends into one of its subloops," whereupon R becomes a sum over *all descent paths* (Corollary 1); **multiplying over child terms = the terms *coexist* (conjunctive/AND)**, corresponding to "all subloops present at once, their scores multiplied." The two differ sharply in their political reading:

$$\underbrace{\textstyle\sum_j \text{ (OR)}}_{\substack{\text{rewards "many regenerative paths":} \\ \text{redundancy, diversity, networks}}} \quad \text{vs.} \quad \underbrace{\textstyle\prod_j \text{ (AND)}}_{\substack{\text{rewards "all subloops healthy at once":} \\ \text{holistic synergy}}} \tag{1.21}$$

Eglash wrote $\sum$, so the equation has *already* selected the OR reading – R measures "how many outer-to-bottom regenerative chains the system supports," not "whether all subloops are simultaneously healthy." But Eglash's *intuitive description* of nested loops (a parent loop "containing" several subloops, seemingly present at once) is closer to AND. This is a genuine tension between the equation as written and the prose intuition. The value of the reduction is to *put it on the table*: if the intent is OR, the present equation is correct, and its "reward many paths" semantics aligns with generative justice's preference for diversity, redundancy, and networks; if the intent is AND, then $\sum$ should be changed to $\prod$, and (1.2) rewritten accordingly. This choice cannot be made *for* Eglash by formalization; it must be settled by Eglash – it bears directly on what R actually rewards. (See, however, the bridge to Claim 2 in Remark 4, which dissolves much of this tension.)

(iii) The redundancy of d: depth is not an independent weight

Remark 5(ii) notes that the standard inside recursion has no independent depth factor, since depth is already implicit in the tree structure. The explicit d factor in R means *depth is counted twice*: once through the recursion r rolling subtrees up layer by layer (which already carries depth information), and once through the multiplicative d. The consequence is that R rewards deep nesting *superlinearly*. This is not necessarily an error – if the theoretical intent is precisely to give an extra reward, beyond structure itself, to "nesting deeper," then d is a deliberate design choice. But it should be made explicit as a *normative weight choice*, not a neutral structural fact. The value of the reduction is exactly to separate this choice from the appearance of the equation, so that it can be discussed and defended on its own.

But a further honest note is needed here, because a tension hides between Remark 5(i) and (ii), and the strength of the "superlinearity" depends precisely on how it is resolved. The original

appendix gives d two incompatible readings: the outer equation reads d_i as a *global* depth ("2 for children, 3 for grandchildren"), while the inner equation reads d_j as a *local* depth ("starting from 1 for the parent loop's direct subloops"). **Under the local reading**, each edge's d_j along a path is constantly 1 (each node is its parent's direct child), so the depth factor does *not* compound along the path, and the previous paragraph's "superlinearity" does not hold – the only nontrivial d is the d_i on the root edge. **Under the global reading**, depth does compound along the path, producing that superlinear (indeed near-factorial) amplification, but then the "local counting" of Remark 5(i) no longer applies. The two cannot both hold: the original equation uses both countings at once, so "what d rewards, and how strongly" is undetermined in the original. This again confirms that d is not a neutral structural quantity but a normative choice *awaiting Eglash's clarification.*

(iv) The dichotomy of parsing semantics: inside vs. Viterbi

Theorem 1 fixes the probability semiring because the original literally uses $\sum$. This endows R with *total inside score* semantics: R is the weighted sum over *all* regenerative derivation paths of the system. If the theoretical intent is really "the system's *deepest/optimal* single regenerative chain," then one should switch to the tropical semiring $(\mathbb{R}\cup\{\infty\}, \min, +)$, and $\beta(S)$ becomes the Viterbi optimal parse. The two semantics differ in political meaning: inside semantics reward "many regenerative paths" (redundancy, diversity, networks), Viterbi semantics reward "the depth of the single best chain." The semiring framework makes this choice explicit, while the original equation's $\sum/\times$ form defaults to the former.

(v) What the reduction lacks: the entry point to the next claim

Finally, the reduction also *illuminates what it cannot contain.* Grammatical parsing (whether PCFG or PCSG) is about *structure*: it scores a given derivation tree. But what makes a PCFG a "probability" grammar – its entire rigidity – comes from one normalization constraint: for each nonterminal A, $\sum_\beta p(A \to \beta) = 1$. This constraint is a *local conservation law*: at each node all outgoing options share a fixed probability mass, with no global legislator prescribing the total scale. The edge weights (1.10) of Eglash's R satisfy *no* such normalization: W is adjustable, P, E have no constrained range, d accumulates arbitrarily. Hence R is an *unnormalized* weighted parse – it gives an absolute accumulated score, not mutually competing relative probabilities.

The significance of this observation goes beyond the technical. A local normalization constraint is, in political semantics, exactly "bottom-up, decentralized conservation": the scale emerges from each local node, rather than being prescribed from the top. An *unnormalized* weighted parse, by contrast, hands the legitimacy of the scale back to an external weight-setter (who tunes W, who fixes P, E). The reduction therefore locates precisely the entry point of the subsequent work: to make R truly a measure of a "social physics," what is missing is not more variables but a *local conservation law* – one that simultaneously answers "by what right is R comparable" and "to whom does the scale belong." This is taken up in Claim 2.

Grammar object	R **symbol**	**Semantic gloss**	**Coh.?**
nonterminal $A \in V$	a loop	a unit further decomposable into subloops	yes
terminal Σ	leaf / atomic regenerative process	indecomposable, $\beta = \mathbf{1}$	yes
M_i unary productions $A \to B_j$	"loop A descends to one of its M_i subloops"	disjunctive (OR) branch; inside *sums* over them (Rem. 2)	yes
nested structure τ (OR tree)	system's nested-loop structure	parent–child = tree edge; a derivation = a root-to-leaf path	yes
root-to-leaf path π	an "outer-to-bottom" regenerative chain	$R = \sum_\pi \prod_{\text{edge}} w$ (Cor. 1)	yes
root S	outermost loop (whole system)	–	yes
inside score $\beta(v)$	r_i (internal recursive depth)	accumulated score rolled up from the subtree	yes
root score $\beta(S)$	R (total recursive depth)	conclusion (1.15) of the reduction theorem	yes
edge weight $w_{i\to j}$	$W_j d_j P_{j-1} E_j$	composite weight of the parent–child junction	yes
$\hookrightarrow P_{j-1}$	response efficacy to the parent loop	on the parent–child edge; *intra-tree* coupling	yes
$\hookrightarrow E_j(\gamma, \delta)$	response efficacy to the environment	weight's dependence on *context* (where PCSG lives)	yes [†]
$\hookrightarrow W_j$	same-level redundancy weight	pure grammar coefficient (semiring weight)	yes
$\hookrightarrow d_j$	depth	no such factor in standard parsing; see Rem. 5	no

Table 1: Symbol-by-symbol semantic mapping from the recursive-depth equation to PCSG parsing. A row marked "$\hookrightarrow$" is a multiplicative sub-factor of the edge weight in the row above. [†] The coherence of E is exactly what requires the *weighting-sense* context-sensitive level (Remark 1): it has no place in a PCFG, while in a PCSG it lands naturally as the weight's context-dependence.

2 Claim 2: Mechanical Conversion of a PCSG into a Dynamical System

2.1 Claim 2: the conversion proposition

Claim 1 shows that Eglash's recursive depth R is a parsing score over a probabilistic context-sensitive grammar (PCSG). This places "generativity" within the framework of *formal languages*. But Eglash's real ambition – and the vocabulary recurring throughout Eglash's book (basin of attraction, trajectory, self-maintenance, criticality) – belongs to the framework of *dynamical systems*. The task of this section is to build a provable bridge between the two frameworks.

Claim 2 (Mechanical conversion from PCSG to a dynamical system)**.** Any probabilistic context-sensitive grammar G can be *mechanically* (i.e. by a definite construction algorithm acting on G's productions and weights) converted into at least one *dynamical-system model*. More precisely: there exists a construction Φ mapping G to a *multitype branching process* $\Phi(G)$, such that the derivations of G agree in distribution with the population evolution of $\Phi(G)$; and the spectral radius $\rho(\mathsf{M}) = \lambda$ of the process's *expectation operator* M gives a dynamical criterion: for $\lambda > 1$ the system expands (self-maintaining, regenerative), for $\lambda < 1$ the system contracts (cannot maintain itself, dies out), and $\lambda = 1$ is critical. Furthermore, this discrete-time process has a well-defined *continuous-time limit*, whose basin-of-attraction structure in phase space connects to Eglash's basin/trajectory vocabulary. One conclusion deserves special emphasis, derivable from this criterion yet invisible to R itself: *recursive depth is not regeneration* – only when the loop network *closes into a cycle* with gain exceeding 1 can the system maintain itself (Proposition 1).

Remark 6 (Why this claim is nontrivial)**.** "Any discrete computing device can be simulated by some dynamical system" is a *trivial* existential fact (a Turing machine can be simulated by ODEs). If Claim 2 asserted only this, it would be worthless. The content of Claim 2 lies *not* in "can be simulated" but in *structure preservation*: the linguistic structure of G (nonterminals, productions, weights) is mapped to *meaningful* dynamical structure (types, branching, spectrum), so that linguistic quantities (such as the regenerative responses E, P) correspond to dynamical quantities (such as survival and growth rates), and G's core dichotomy (regeneration/extraction) corresponds to a *spectral criterion* ($\lambda \gtrless 1$). This is isomorphic to the nontriviality of Claim 1: the value is not in "can be reduced/converted" but in "every concept remains semantically coherent after the mapping."

Structure of this section: definition of the target class of dynamical systems (§2.2); the construction of the mechanical conversion (§2.3); the distributional-consistency theorem (§2.4); the spectral-criterion theorem (§2.5); the continuous-time limit and basins of attraction (§2.6); the conceptual mapping (§2.7); discussion (§2.8).

2.2 Definition of the target class of dynamical systems

Definition 5 (Multitype branching process)**.** Let there be a finite type set $\mathcal{T} = \{1, \ldots, K\}$. A *multitype branching process* is a discrete-time Markov chain $\{\mathbf{Z}_t\}_{t\in\mathbb{N}}$ whose state $\mathbf{Z}_t = (Z_t^{(1)}, \ldots, Z_t^{(K)}) \in \mathbb{N}^K$ records the number of individuals of each type in generation t. Each individual of type i independently produces, in one step, a set of offspring (a multiset over $\mathbb{N}^K$) according to an *offspring distribution* p_i, with different individuals and different generations mutually independent. The evolution is completely determined by the p_i.

Definition 6 (Expectation operator)**.** The *expectation operator* (a.k.a. mean matrix) of a branching process is the $K \times K$ nonnegative matrix $\mathsf{M} = (m_{ij})$, where

$$m_{ij} \;=\; \mathbb{E}[\text{number of type-}j\text{ offspring produced in one step by a type-}i\text{ individual}]. \tag{2.1}$$

M governs the linear evolution of the expected population: $\mathbb{E}[\mathbf{Z}_{t+1} \mid \mathbf{Z}_t] = \mathbf{Z}_t\,\mathsf{M}$, hence $\mathbb{E}[\mathbf{Z}_t] = \mathbf{Z}_0\,\mathsf{M}^t$.

Definition 7 (Spectral radius and regeneration criterion)**.** Write the *spectral radius* of M as $\lambda = \rho(\mathsf{M}) = \max\{|\mu| : \mu \in \operatorname{spec}(\mathsf{M})\}$. When M is irreducible, by the Perron–Frobenius theorem [7] $\lambda > 0$ is a simple real eigenvalue with positive left and right eigenvectors. Call

$$\lambda > 1 \text{ (supercritical, expanding)}, \qquad \lambda = 1 \text{ (critical)}, \qquad \lambda < 1 \text{ (subcritical, dying out)} \tag{2.2}$$

the *regeneration criterion* of this dynamical system.

Remark 7. These are standard objects of branching-process theory (Harris [4]; Athreya–Ney [5]). This section invents no new mathematics, but uses them as the *codomain* of the conversion Φ. The entire work of Claim 2 lies in constructing Φ (§2.3) and proving structure preservation (§2.4–2.5).

2.3 Construction of the mechanical conversion Φ

Let the PCSG be $G = (V, \Sigma, \mathcal{R}, S, \mathbb{K}, w)$ (using the definitions of Claim 1; the semiring is the probability semiring $\mathbb{K} = (\mathbb{R}_{\geq 0}, +, \times, 0, 1)$). Construct $\Phi(G)$ as follows.

Step 1: type set. Let the process's type set be the set of nonterminals, $\mathcal{T} := V$, $K := |V|$. The terminals Σ are regarded as "individuals that no longer split" (types of zero fertility), corresponding to leaf loops.

Step 2: two operators – fertility and shape. For each nonterminal $A \in V$ with production set $\{A \to \alpha_1, \ldots, A \to \alpha_n\}$ and (context-sensitive) weights $w(A \to \alpha_k \mid \gamma, \delta)$, *two* non-conflatable operators can be read off the *same* weights (Remark 9):

- **Fertility operator (unnormalized, carries growth).** Read the edge weight directly as an *expected number of offspring*: a type-A individual produces, in one step, an expected number of "each nonterminal in α_k" of offspring given by $w(A \to \alpha_k \mid \gamma, \delta)$ (e.g. let each offspring count be independently Poisson with mean the corresponding edge weight). This gives the fertility offspring law p_A^{fert}, whose mean-matrix row sums may be greater than, equal to, or less than 1, and which therefore *carries genuine growth information.*
- **Shape distribution (normalized, carries structural probability).** If one only asks "into which structure does A expand," normalize the weights into probabilities

$$\pi(A \to \alpha_k \mid \gamma, \delta) \;=\; \frac{w(A \to \alpha_k \mid \gamma, \delta)}{\sum_{l=1}^{n} w(A \to \alpha_l \mid \gamma, \delta)}, \tag{2.3}$$

  giving the offspring *shape* law p_A^{shape}: choose production α_k with probability π. It gives G's probability measure over all derivation trees (Theorem 2), but *discards growth* – each individual expands by exactly one production, so it answers only "reproduce to *where*," not "reproduce how *much*."

The key distinction (Remark 9): the regeneration criterion λ lives in the *fertility* operator. The mean matrix of the shape distribution, when the productions are unary (the OR reading of Claim 1), is a *stochastic matrix* with spectral radius identically 1, carrying no information about growth. Hence the spectral criterion of §2.5 below is built throughout on the fertility operator; normalization is used only to characterize the probability of the "shape" (Theorem 2).

Step 3: handling the context. The context (γ, δ) makes p_A depend on the environment, so $\Phi(G)$ is in general a *context-dependent* branching process – that is, by attaching a finite "context label" to V, lift $\Phi(G)$ to a *finite-type-extended* branching process $\widehat{\Phi}(G)$ with type set $\widehat{\mathcal{T}} = V \times \mathrm{Ctx}_{\mathrm{fin}}$, where $\mathrm{Ctx}_{\mathrm{fin}}$ is the finite set of local contexts actually occurring in the derivation. Over $\widehat{\mathcal{T}}$, the offspring distribution no longer depends on an external context (the context has been encoded into the type), so $\widehat{\Phi}(G)$ is a standard multitype branching process in the sense of Definition 5.

Remark 8 (Finiteness of context absorption)**.** Step 3 is feasible provided only finitely many local contexts actually occur in any derivation. For the *branching derivations* of Claim 1 (each step rewriting a single nonterminal), this finiteness holds: each node's local context is determined by its parent and sibling labels in the derivation tree, which are drawn from the finite set $V \cup \Sigma$. Hence $\widehat{\mathcal{T}}$ is finite and $\widehat{\Phi}(G)$ is well-defined. This is exactly the payoff in Claim 2 of Claim 1's "branching derivation" assumption: it simultaneously guarantees the tree structure (Claim 1) and the finiteness of the type set (Claim 2).

Step 4: fertility (expectation) operator. By (2.1), the entries of the fertility operator $\mathsf{M} = (m_{(A,c),(B,c')})$ of $\widehat{\Phi}(G)$ are the expected number of type-(B, c') offspring of a type-(A, c) individual, *read directly off the unnormalized weights*:

$$m_{(A,c),(B,c')} \;=\; \sum_{A \to \alpha \,|\, c} w(A \to \alpha \mid c) \;\cdot\; \#\{\, B \text{ occurs in } \alpha \text{ with inherited context } c' \,\}, \qquad (2.4)$$

where $\#\{\cdot\}$ is multiplicity counting. Matching the edge-weighting of Claim 1, the entries of M are exactly the edge weights $m_{i\to j} = w_{i\to j} = W_j d_j P_{j-1} E_j$. M is read mechanically off G's productions and weights, with *no* normalization – the growth information lies precisely in the unnormalized row sums.

Thus $\Phi : G \mapsto \widehat{\Phi}(G)$ is a construction mechanically determined by G, with codomain the multitype branching processes (Definition 5). □

Remark 9 (Fertility vs. shape: two operators, one carries λ)**.** Please keep the two operators of Step 2 distinct, or the spectral criterion fails. **(a) Fertility operator** $\mathsf{M} = (w_{i\to j})$ ((2.4), unnormalized): the row sums $\sum_j w_{i\to j}$ are free, the spectral radius $\lambda = \rho(\mathsf{M})$ can be $\gtrless 1$, and it *carries growth*; the criterion of §2.5 is built entirely on it. **(b) Shape (stochastic) operator** $\Pi = (\pi_{i\to j})$ ((2.3), row-normalized): the row sums are identically 1, it is a stochastic matrix, its *spectral radius is identically* 1 (Perron), and it only characterizes the probability of derivation trees (Theorem 2), carrying no growth information. Their division of labor is exactly that of §2.8, item (ii): "reproduce to where (Π: conserved, local)" vs. "reproduce how much (M: can be supercritical, enters λ)." **If one mistakenly builds the spectral criterion on Π, then $\lambda \equiv 1$ and the criterion becomes vacuous** – this is exactly the dynamical echo of Claim 1's missing normalization (the missing conservation law).

2.4 Distributional-consistency theorem

For the conversion to "preserve structure," the most basic requirement is that the random mechanism by which G generates derivations and the random mechanism by which $\widehat{\Phi}(G)$ breeds populations be the *same* probability distribution. The following theorem establishes this.

Theorem 2 (Derivation–population distributional consistency)**.** *Let G be a PCSG, $\widehat{\Phi}(G)$ the branching process constructed in §2.3, with initial population a single individual of type $(S, \varnothing)$ (corresponding to the start symbol). Then G's random derivation tree (productions drawn by the normalized* shape *probability $\pi = p^{\text{shape}}$; this distribution characterizes structure only and is independent of the growth-carrying fertility operator, see Remark 9) and the population family tree of $\widehat{\Phi}(G)$ are equal in distribution: for any finite tree skeleton τ,*

$$\Pr_G[\tau] \;=\; \Pr_{\widehat{\Phi}(G)}[\tau] \;=\; \prod_{v \in \mathcal{N}(\tau)} \pi\big(\mathit{rule}(v) \mid \mathrm{ctx}(v)\big). \tag{2.5}$$

Proof. Both distributions are product measures of "independently choosing an expansion/offspring per node by π," so it suffices to verify that the per-node local mechanisms agree. On the G side: a node v (label A, context c) expands to α with probability $\pi(A \to \alpha \mid c)$ by (2.3), and the choices at different nodes are conditionally independent (branching derivations make the rewriting steps non-interfering). On the $\widehat{\Phi}(G)$ side: a type-(A, c) individual independently produces offspring α by the offspring distribution $p_{(A,c)}$ (the same π) (Steps 2–3). The local mechanisms agree pointwise and the independence structures coincide, so the product measures are equal, giving (2.5). □

Corollary 3 (Claim 1's R is the expected total progeny (Green's function))**.** *Claim 1's recursive depth $R = \beta(S)$ is exactly the* expected-total-progeny functional *of the fertility process $\widehat{\Phi}(G)$. Indeed, Claim 1 already proved R equals the path-sum $\sum_{\pi: S \to \mathit{leaf}} \prod_{(i \to j) \in \pi} w_{i \to j}$ (the "path-sum closed form" corollary of Claim 1), and with $m_{i \to j} = w_{i \to j}$ as the fertility matrix,*

$$R \;=\; \sum_{\pi: S \to \mathit{leaf}} \; \prod_{(i \to j) \in \pi} m_{i \to j} \;=\; \mathbf{e}_S\Big(\textstyle\sum_{t \geq 0} \mathsf{M}^t\Big)\mathbf{1} \;=\; \mathbf{e}_S (I - \mathsf{M})^{-1}\mathbf{1} \quad (\textit{when } \lambda < 1), \tag{2.6}$$

i.e. the sum over all generations of expected progeny starting from the root S (the Green's function $(I - \mathsf{M})^{-1}$). A finite nesting is the truncation of this series at finite depth.

Proof. The path-sum identity is from Claim 1. In the branching process, starting from a single S, the expected number of each type in generation t is $\mathbf{e}_S \mathsf{M}^t$; summing over all generations gives $\mathbf{e}_S \sum_{t \geq 0} \mathsf{M}^t = \mathbf{e}_S (I - \mathsf{M})^{-1}$ (the series converges when $\lambda = \rho(\mathsf{M}) < 1$). The (S, j) entry of M^t is exactly the sum of edge-weight products over length-t root-to-j paths, so the series is the path expansion $\sum_\pi \prod w$. □

Remark 10 (R finite ⇔ subcritical)**.** Equation (2.6) exposes a fact invisible to Claim 1: under infinite nesting, R converges *if and only if* $\lambda < 1$. In other words, Claim 1's R is meaningful as a finite number precisely when the system *cannot* self-maintain supercritically ($\lambda < 1$); once $\lambda \geq 1$, R diverges and one must use the growth rate λ itself as the measure. This pins down the division of labor between R and λ: R is a finite accumulated quantity in the subcritical regime, λ is the universal cross-critical criterion.

2.5 Spectral-criterion theorem: regeneration is spectrum, depth is not regeneration

This section is the core of Claim 2. It proceeds in two steps: first the *spectral characterization* of regeneration/extraction (Theorem 3), then a conclusion crucial for generative justice yet hidden in R – **recursive depth is not regeneration** (Proposition 1). The latter is the real payoff of this "grammar → dynamics" bridge.

Theorem 3 (Spectral characterization of regeneration/extraction)**.** *Suppose $\widehat{\Phi}(G)$ is irreducible (any type can produce any type in finitely many steps; equivalently, G's production graph is strongly connected). Write $\lambda = \rho(\mathsf{M})$. Then:*

*(i) (**extinction / extraction**) If $\lambda < 1$, the population goes extinct with probability 1: $\Pr[\exists t : \mathbf{Z}_t = \mathbf{0}] = 1$. The expected population decays exponentially as $\lambda^t \to 0$.*

*(ii) (**expansion / regeneration**) If $\lambda > 1$, the population grows unboundedly with positive probability, and on the non-extinction event $\mathbf{Z}_t/\lambda^t$ converges to a positive random limit determined by the Perron eigenvector (Kesten–Stigum [6]).*

*(iii) (**critical**) If $\lambda = 1$ (and the offspring distribution is nondegenerate), the population goes extinct with probability 1, but the expected extinction time diverges – the system sits at the edge of self-maintenance.*

Proof. These are classical results of multitype branching processes (Harris' theorem; Athreya–Ney, Ch. V). (i),(iii) follow from $\mathbb{E}[\mathbf{Z}_t] = \mathbf{Z}_0\mathsf{M}^t$ and Perron–Frobenius ($\mathsf{M}^t \sim \lambda^t\, vu^\top$) giving the exponential behavior of the expectation, together with the classification of solutions of the extinction-probability fixed-point equation $\mathbf{q} = \mathbf{f}(\mathbf{q})$; (ii) is the Kesten–Stigum theorem (under the $L \log L$ moment condition). M is given mechanically by (2.4), so λ is mechanically determined by G. □

Corollary 4 (Dynamical translation of Eglash's dichotomy)**.** *Under the conversion Φ, Eglash's core dichotomy acquires a* spectral criterion*:*

$$\underbrace{\textit{self-sustaining}}_{\lambda > 1} \;\Big|\; \underbrace{\textit{critical (knife-edge)}}_{\lambda = 1} \;\Big|\; \underbrace{\textit{extractive (regeneration impaired)}}_{\lambda < 1}. \tag{2.7}$$

The strict *inequality $\lambda > 1$ here is necessary: by Theorem 3(iii), a critical population $\lambda = 1$ still goes extinct with probability 1 (only the expected extinction time diverges), so genuine self-maintenance requires supercriticality $\lambda > 1$, while $\lambda = 1$ is the knife-edge that "dies out eventually, but infinitely slowly." "Extraction" is, dynamically, "the spectral radius of the fertility operator is pressed below 1, so the system cannot sustain itself." The response efficacies E, P enter the matrix entries* (2.4) *of M: any E or P tending to 0 multiplicatively lowers the corresponding entry and hence lowers λ. This agrees, at the dynamical level, with the "multiplicative veto" semantics of E, P in Claim 1.*

Core conclusion: depth is not regeneration. The spectral criterion immediately forces a fact invisible to R.

Proposition 1 (Acyclic $\Rightarrow \lambda = 0$: recursive depth is not regeneration)**.** *Suppose the* type graph $\mathcal{G}_{\mathsf{M}}$ *of the fertility matrix* M *(an edge $i \to j$ whenever $m_{ij} > 0$) is* acyclic *(a DAG) – this is exactly the case of "finitely many layers of nesting, where subloops do not contain their ancestor loops." Then* M *is nilpotent,* $\mathrm{spec}(\mathsf{M}) = \{0\}$*, so*

$$\lambda = 0 < 1: \quad \textit{the system is subcritical, with finite expected total progeny and almost-sure extinction.} \tag{2.8}$$

More generally,

$$\lambda > 0 \iff \mathcal{G}_{\mathsf{M}} \textit{ contains a directed cycle (some loop regenerates its own type through its descendants)}, \tag{2.9}$$

and self-maintenance $\lambda > 1$ *further requires at least one regenerative cycle whose* net gain *(the product of edge weights along the cycle) exceeds* 1*.*

Proof. For an acyclic graph, in some topological order the (weighted) adjacency matrix is strictly upper triangular, so M is nilpotent, all eigenvalues are 0, and $\lambda = \rho(\mathsf{M}) = 0$; subcriticality follows from Theorem 3(i). In general, $\lambda = \rho(\mathsf{M}) > 0$ iff M is not nilpotent, iff there exists k with $\operatorname{tr} \mathsf{M}^k > 0$, i.e. the type graph contains a directed closed walk of length k; this gives (2.9). The product of edge weights along a cycle is that cycle's contribution to a principal submatrix of M, and when it exceeds 1 one has $\rho > 1$. □

Remark 11 (The separation of "recursive depth" from "regeneration" – the sharpest payoff of this bridge)**.** Proposition 1 nails down a distinction hidden in R. Eglash's R rewards *depth of nesting* (and, through the d factor, rewards it *superlinearly*, see the Claim 1 remarks); but a deep yet *acyclic* structure has $\lambda = 0$ – it eventually dies and does not self-maintain at all. **Genuine regeneration lies not in "how deep the nesting," but in "whether it closes into a cycle": a loop must *regenerate itself* through its descendants for λ to be positive; its net gain must exceed 1 for the system to self-maintain.** Thus:

$$\underbrace{\text{deep, acyclic, extractive assembly line}}_{R \text{ large, } \lambda=0} \qquad \text{vs.} \qquad \underbrace{\text{shallow, closed-loop regenerative cycle}}_{R \text{ small, } \lambda>1}. \tag{2.10}$$

By R the former wins; by λ only the latter is generative. This *formalizes* exactly Eglash's own critique of extractive AI as having "large assembly depth but shallow recursive depth": assembly depth is path length in a DAG (measured by R), whereas the true meaning of regeneration is the existence of a *cycle* and its gain (measured by λ). λ is therefore not merely the asymptotic rate of R, but a *correction* to R – it penalizes structures that disguise depth as regeneration.

2.6 Continuous-time limit and basins of attraction

Eglash's vocabulary (basin of attraction, trajectory, region) belongs to *continuous-time* dynamical systems. The discrete branching process $\widehat{\Phi}(G)$ has a standard continuous-time limit, making this connection possible.

Proposition 2 (Expectation flow and fixed points)**.** *Writing the expected density of each type as* $\mathbf{x}(t) \in \mathbb{R}^K_{\geq 0}$ *(here* M *is the* fertility *operator, $\rho(\mathsf{M}) = \lambda$), the continuous-time limit of the expected evolution of $\widehat{\Phi}(G)$ is the linear flow*

$$\dot{\mathbf{x}} \;=\; (\mathsf{M} - I)\,\mathbf{x}\,, \tag{2.11}$$

whose phase-space behavior is governed by the spectrum of $\mathsf{M} - I$*: for* $\lambda > 1$ *(i.e.* $\mathsf{M} - I$ *has a positive eigenvalue) the origin is a* repelling *fixed point and the density expands exponentially along the Perron direction; for* $\lambda < 1$ *the origin is an* attracting *fixed point (with basin of attraction the entire positive orthant) and the density collapses to zero. Nonlinear offspring interactions (resource constraints, density dependence) generalize* (2.11) *to*

$$\dot{\mathbf{x}} \;=\; \mathbf{F}(\mathbf{x}), \qquad \mathbf{F}(\mathbf{x}) = (\mathsf{M}(\mathbf{x}) - I)\,\mathbf{x}, \tag{2.12}$$

where the density dependence of $\mathsf{M}(\mathbf{x})$ *produces nontrivial* basins of attraction – *the formal counterpart of Eglash's "basin of attraction."*

Proof. Equation (2.11) is the standard continuization of the discrete map $\mathbf{x}_{t+1} = \mathsf{M}\mathbf{x}_t$ (small-step limit $\mathbf{x}_{t+1} - \mathbf{x}_t \approx \dot{\mathbf{x}}$). The stability of the fixed point $\mathbf{x}^* = \mathbf{0}$ is determined by the sign of the eigenvalues of $\mathsf{M} - I$, and whether $\rho(\mathsf{M}) = \lambda$ is greater or less than 1 exactly determines whether $\mathsf{M} - I$ has a positive eigenvalue. Equation (2.12) is the standard density-dependent generalization, whose basin structure is given by the fixed points of $\mathbf{F}$ and the spectra of their linearizations. □

Remark 12 (Connecting to Eglash's vocabulary)**.** Proposition 2 nails Eglash's three metaphors into mathematical objects: *trajectory* = a solution curve of the flow (2.12); *basin of attraction* = the set of initial conditions converging to a given stable fixed point; *self-maintenance* = the existence of a nonzero stable fixed point (a critical equilibrium where $\lambda \geq 1$ at that point). Eglash's "niche alteration" (modifying one's own basin) corresponds to the density dependence $\mathsf{M}(\mathbf{x})$: the system's state feeds back on its own expectation operator, making the basin vary along the trajectory. This will be developed in Claim 3 (background independence).

2.7 Conceptual mapping

The overall picture of the mapping: **all of the PCSG's generative structure (types, productions, weights, context) lands mechanically in the corresponding structure of the branching dynamical system; and the conversion additionally *produces* a quantity absent on the linguistic side – the spectral radius** λ**.** λ is the real payoff of this bridge: it lifts "generativity" from a static score (R) to a *dynamical growth rate*, and turns "regeneration/extraction" from a qualitative judgment into a *critical criterion* ($\lambda \gtrless 1$).

2.8 Discussion and analysis

(i) The relation between R and λ: a static score as a proxy for the growth rate

Claim 1's R and Claim 2's λ do not measure the same thing, but both are generated by the same set of weights $\{W, d, P, E\}$, so there is a precise link. $R = \beta(S)$ is the path-sum of the expectation operator M over a *finite-depth truncation* (Corollary 3); whereas $\lambda = \rho(\mathsf{M})$ is the asymptotic growth rate of M^t as $t \to \infty$. By Perron–Frobenius, $\mathbf{e}_S \mathsf{M}^t \mathbf{1} \sim C\lambda^t$, so *the growth rate of the depth-truncated* R *with truncation depth* T *is exactly* λ:

$$\lambda \;=\; \lim_{T\to\infty} \frac{R_{T+1}}{R_T}\,, \tag{2.13}$$

where R_T is the value of the inside recursion truncated to depth T. In other words: **Eglash's** R **is a finite-depth proxy for the dynamical growth rate** λ**;** λ **is the asymptotic rate**

PCSG (linguistic)	Branching dynamical system	Semantic correspondence
nonterminal $A \in V$	individual type $i \in \mathcal{T}$	a class of reproducible loop
terminal Σ	type of zero fertility	leaf / atomic process
production $A \to \alpha$	offspring event	"A produces the offspring multiset α"
unnorm. edge weight $W_j d_j P_{j-1} E_j$	fertility-matrix entry m_{ij}	Eq. (2.4); carries growth
normalized weight π	shape distribution p^{shape} (stochastic, $\rho{=}1$)	Eq. (2.3); structure only (Rem. 9)
derivation tree τ	population family tree	Thm. 2: equal in distribution
inside score R	expected total progeny $\mathbf{e}_S(I - \mathsf{M})^{-1}\mathbf{1}$	Cor. 3; $R < \infty \Leftrightarrow \lambda < 1$
$\hookrightarrow P, E$ (response efficacy)	multiplicative factors of m_{ij}	$E, P \to 0 \Rightarrow m_{ij} \to 0 \Rightarrow \lambda \downarrow$
context (γ, δ)	type's context label	finite-type extension (Rem. 8)
–	spectral radius $\lambda = \rho(\mathsf{M})$	**regeneration criterion** ($\lambda \gtrless 1$); no preimage on the R side
regeneration / extraction	$\lambda > 1$ / $\lambda < 1$ ($\lambda{=}1$ critical)	Cor. 4
recursive depth $\neq$ regeneration	acyclic $\Rightarrow \lambda = 0$; regeneration needs a *cycle*	**Prop. 1**
basin / trajectory (metaphor)	basin / solution curve of (2.12)	Prop. 2

Table 2: Item-by-item semantic mapping from PCSG to the multitype branching dynamical system. A row marked "$\hookrightarrow$" is a sub-factor of the row above. The most salient item: λ has *no* preimage on the linguistic side – it is a dynamical quantity *newly produced* by the conversion, lifting a static structural score into a dynamical growth-rate criterion.

of R at infinite nesting depth. This gives R a dynamical standing it previously lacked – it is not an arbitrary score but a truncated estimate of a growth rate. It also explains why R is sensitive to depth (the superlinearity problem of d in the Claim 1 remarks): the deeper the truncation, the closer to λ.

(ii) Normalization: the conservation law missing from Claim 1, forced to surface here

The discussion of Claim 1 noted that R lacks a local normalization constraint. The conversion Φ *forces* this constraint to surface: to read the weights as *probability* offspring distributions, Step 2 (2.3) must normalize for each nonterminal, $\sum_k \pi(A \to \alpha_k) = 1$. This is exactly the conservation law that makes a PCFG/PCSG a "probability" grammar – each type's reproduction probabilities share a fixed mass. Its dynamical meaning is: **normalization separates "reproduce to where" from "reproduce how much"** – the former a probability (conserved, local, bottom-up), the latter a fertility (can be supercritical, enters λ). Eglash's R mixes the two in one unnormalized accumulation, so it is neither a probability nor a pure growth rate; the conversion

forces us to split them apart, and only after splitting does λ emerge.

(iii) Whose criterion: λ and the "who" question

In Eglash's correspondence, Eglash lists "who reaps, who controls the conditions" (the "who") as the third question of generative justice, with Democritus' "bottom-up" against Plato's "philosopher-kings." The spectral criterion λ is itself *structural* (regeneration vs. extraction) and does not yet directly encode "who." But the conversion reveals the precise entry point of "who": λ is determined by the *spectrum of the expectation operator*, and the spectrum is determined by the connectivity of the *whole type network* (the irreducibility hypothesis of Theorem 3). "Who" (the distribution of control) corresponds to *who can change the entries of* M – i.e. who controls the production weights w. In a top-down structure, the entries of a few types (philosopher-kings) dominate λ; in a bottom-up structure, λ is determined by the decentralized connectivity of the whole network (the Perron vector spread uniformly over the types). **Thus the "who" question is translated into: the concentration of the Perron eigenvector – whether it concentrates on a few types (Plato) or spreads uniformly over the whole network (Democritus).** This nails a political question into a spectral concentration measure (e.g. the entropy or participation ratio of the Perron vector), to be developed systematically in Claim 3.

(iv) Non-uniqueness of the conversion: the meaning of "at least one"

Claim 2 asserts "at least one" dynamical system, not a unique one. This is honest and necessary: the same PCSG can be converted into (a) a discrete branching process (the main line of this section), (b) the iteration of its expectation matrix (a deterministic linear map), (c) the continuous-time expectation flow (2.11), (d) the density-dependent nonlinear flow (2.12), and even (e) a local update rule on a lattice (connecting to Democritus' atomist imagery). These are different *dynamical realizations* of the same grammar, sharing the same M and the same λ, but differing in randomness, time-continuity, and spatial structure. "At least one" is therefore not a weakening but a point: λ **is the invariant of the conversion** – whichever realization is chosen, the regeneration criterion $\lambda \gtrless 1$ is unchanged. This invariance is the real hard core of Claim 2: it makes "the regenerativity of a PCSG" a well-defined quantity independent of the details of the dynamical realization.

Acknowledgements

I am deeply grateful to Dr. Ron Eglash. A fortunate chance encounter gave me the opportunity to speak with him, and our continuing exchanges about the recursive-depth equation of his theory have been a profound source of inspiration for this work. My own thinking has been deeply shaped by his framework of generative justice.

References

[1] R. Eglash, K. Robinson, R. Robert, M. Nayebare, and A. Bennett. A General Theory of Generativity. *Complexity* (forthcoming), 2025.

[2] J. Goodman. Semiring parsing. *Computational Linguistics*, 25(4):573–605, 1999.

[3] N. Chomsky. On certain formal properties of grammars. *Information and Control*, 2(2):137–167, 1959.

[4] T. E. Harris. *The Theory of Branching Processes*. Springer, Berlin, 1963.

[5] K. B. Athreya and P. E. Ney. *Branching Processes*. Springer, New York, 1972.

[6] H. Kesten and B. P. Stigum. A limit theorem for multidimensional Galton–Watson processes. *Annals of Mathematical Statistics*, 37(5):1211–1223, 1966.

[7] E. Seneta. *Non-negative Matrices and Markov Chains*. Springer, New York, 2nd edition, 2006.