

EXACT DETERMINANT FORMULAS FOR COALESCING PARTICLE SYSTEMS

PIOTR ŚNIADY AND ÁKOS URBÁN

ABSTRACT. When particles on a line collide, they may coalesce into one. Such systems arise in the voter model, where boundaries between opinion clusters perform coalescing random walks, and in reaction-diffusion theory, where diffusing particles merge on contact. Computing exact coalescence probabilities has been difficult because collisions reduce the particle count, while classical determinantal methods require a fixed number of particles throughout. We introduce *ghost particles*: when two particles collide, one survivor continues as usual and one invisible ghost is created alongside it, preserving the total count. This restores the square matrix structure needed for a determinantal formula. We prove that the probability of any specified coalescence pattern—which initial particles merge into which survivors—is given by a determinant whose entries are transition probabilities. Integrating out ghost positions yields a closed-form formula for the surviving particles alone: the *coalescence determinant*. The only assumptions are the Markov property and nearest-neighbor transitions, so the results apply wherever the classical non-colliding theory does: discrete lattice paths, birth-death chains, and continuous diffusions including Brownian motion.

To Marek Bożejko, whose path crossed ours at just the right moments.

1. INTRODUCTION

1.1. The problem. Consider n particles performing independent random walks on \mathbb{Z} . When two particles meet, they coalesce into one. *What is the probability that the particles end up at specified positions, having undergone a specified pattern of collisions?*

This model appears throughout probability and statistical physics, notably in the voter model [HL75]—where tracing ancestry backward in time produces coalescing lineages.

For *non-colliding* particles, exact probabilities are classical. The Karlin–McGregor theorem [KM59] expresses the probability that n particles starting at positions $x_1 < \dots < x_n$ *avoid collision* as a determinant:

$$\Pr(\text{particles reach } y_1, \dots, y_n \text{ without colliding}) = \det(p(x_i \rightarrow y_j))_{1 \leq i, j \leq n},$$

Date: April 8, 2026.

2020 *Mathematics Subject Classification.* Primary 05A15; Secondary 05A19, 15A15, 60C05, 60J65, 82C22.

Key words and phrases. coalescing random walks, ghost particles, Lindström–Gessel–Viennot lemma, Karlin–McGregor formula, determinant, lattice paths, interacting particle systems.

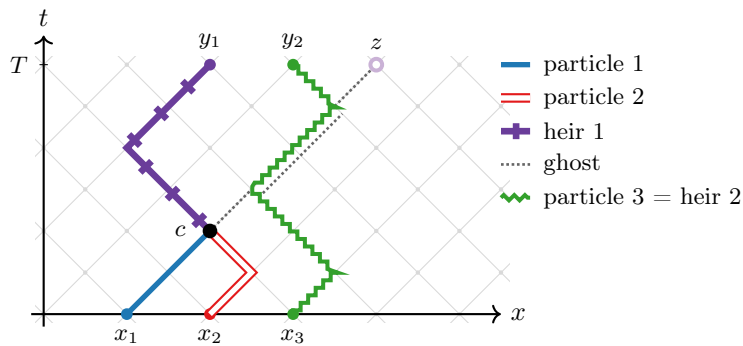


Figure 1. **Coalescence on the checkerboard lattice (pattern 2+1).** Three particles start at $x_1 < x_2 < x_3$. Particles 1 (solid) and 2 (double) coalesce at c , producing heir 1 (ticked) and a ghost (dotted). Particle 3 (zigzag) does not coalesce; it is heir 2. The ghost shares two edges with heir 2 (shown offset)—ghosts do not interact and may cross any path freely. Final positions: $y_1 < y_2 < z$.

where $p(x_i \rightarrow y_j)$ is the transition probability from x_i to y_j for a single particle. The combinatorial version is the Lindström–Gessel–Viennot (LGV) lemma [Lin73; GV85].

When particles coalesce, the count changes. After m coalescences, only $k = n - m$ particles remain. The determinant formula breaks down: the number of rows (initial particles) exceeds the number of columns (final particles), leaving no square matrix.

1.2. The ghost method. Our solution is to keep the discarded particles walking. When two particles merge, the collision produces one *heir* and one *ghost*: the heir is the surviving particle; the ghost is an independent random walk that starts at the collision point and no longer interacts with anything.

The total count—heirs plus ghosts—remains exactly n . With n initial particles and n final entities, we can write a square matrix and take its determinant. The ghosts are not physically real from the viewpoint of coalescing random walks, but the enlarged system admits exact determinantal formulas. We call this the *ghost particle method*.

Figure 1 illustrates this viewpoint for the coalescence pattern 2+1: particles 1 and 2 collide, producing one heir and one ghost, while particle 3 continues independently.

1.3. Main results.

1.3.1. The coalescence formula. Consider n particles starting at positions $x_1 \leq \dots \leq x_n$. Suppose they coalesce into k heirs at positions y_{H_1}, \dots, y_{H_k} , producing $m = n - k$ ghosts at positions y_{g_1}, \dots, y_{g_m} . (We often abbreviate y_{H_i} to H_i and y_{g_j} to g_j , identifying final entities with their positions.) The *coalescence pattern* is described by a *composition* $c_1 + \dots + c_k = n$, where c_j counts how many initial particles merge into heir H_j . Each ghost has a *sign* $\varepsilon(g) \in \{+1, -1\}$: positive if the ghost ends up to the left of its heir, negative if to the right.

Theorem 1.1 (Coalescence formula). *The probability of this outcome is:*

$$\Pr = \left[\prod_g t_g^{\varepsilon(g)} \right] \det(M).$$

The notation $[\prod_g t_g^{\varepsilon(g)}]$ means: expand the determinant in formal variables t_g^+ and t_g^- , then extract the coefficient where each ghost g contributes t_g^+ if $\varepsilon(g) = +1$, or t_g^- if $\varepsilon(g) = -1$.

The $n \times n$ matrix M has rows indexed by initial particles and columns indexed by final entities (heirs and ghosts listed in *rank order*). Heir columns contain the usual transition probabilities $p(x_i \rightarrow y_H)$. Ghost columns have a “staircase” structure with formal variables:

$$M_{i,g} = \begin{cases} t_g^+ \cdot p(x_i \rightarrow g) & \text{if } i \geq \text{rank}(g), \\ -t_g^- \cdot p(x_i \rightarrow g) & \text{if } i < \text{rank}(g). \end{cases}$$

Here $\text{rank}(g)$ is the ghost’s index in the labeling scheme (see Section 2). Figure 2 illustrates the staircase pattern for a larger example.

1.3.2. *Example: coalescence pattern 2+1.* Returning to Figure 1: three particles start at $x_1 < x_2 < x_3$. Particles 1 and 2 collide and merge into heir H_1 ; particle 3 does not collide and continues as heir H_2 . The result: two heirs at positions $H_1 < H_2$ and one ghost at g .

The 3×3 matrix is:

$$M = \begin{pmatrix} p(x_1 \rightarrow H_1) & -t^- p(x_1 \rightarrow g) & p(x_1 \rightarrow H_2) \\ p(x_2 \rightarrow H_1) & t^+ p(x_2 \rightarrow g) & p(x_2 \rightarrow H_2) \\ p(x_3 \rightarrow H_1) & t^+ p(x_3 \rightarrow g) & p(x_3 \rightarrow H_2) \end{pmatrix}.$$

The heir columns (first and third) contain plain transition probabilities. The ghost column (second) multiplies each transition probability by a formal variable: $-t^-$ in the first row and t^+ in the remaining rows—the staircase pattern.

Extracting $[t^+]$ (ghost left of heir, $g \leq H_1$) selects the t^+ entries; cofactor expansion along the ghost column gives two terms:

$$(1.1) \quad \begin{aligned} Z &= p(x_2 \rightarrow g) \cdot \det \begin{pmatrix} p(x_1 \rightarrow H_1) & p(x_1 \rightarrow H_2) \\ p(x_3 \rightarrow H_1) & p(x_3 \rightarrow H_2) \end{pmatrix} \\ &\quad - p(x_3 \rightarrow g) \cdot \det \begin{pmatrix} p(x_1 \rightarrow H_1) & p(x_1 \rightarrow H_2) \\ p(x_2 \rightarrow H_1) & p(x_2 \rightarrow H_2) \end{pmatrix}. \end{aligned}$$

Extracting $[t^-]$ (ghost right of heir, $g > H_1$) selects the single t^- entry; only one term survives:

$$(1.2) \quad Z = p(x_1 \rightarrow g) \cdot \det \begin{pmatrix} p(x_2 \rightarrow H_1) & p(x_2 \rightarrow H_2) \\ p(x_3 \rightarrow H_1) & p(x_3 \rightarrow H_2) \end{pmatrix}.$$

This asymmetry—two terms versus one—reflects the staircase structure: the ghost’s rank determines where the sign changes, and hence how many rows contribute each variable. Such asymmetry is a general feature of the ghost formula.

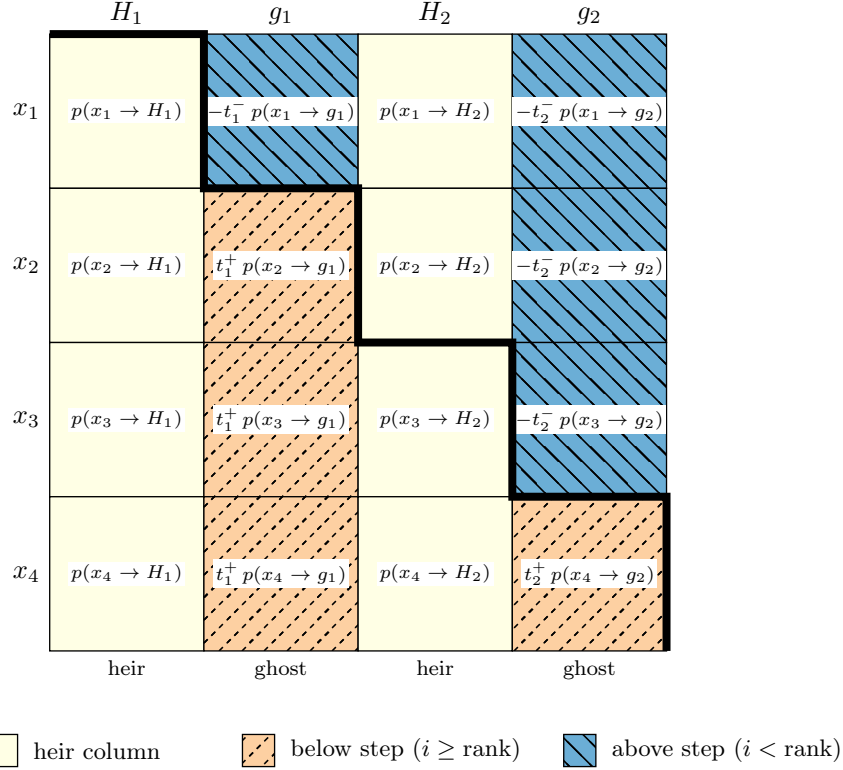


Figure 2. **The coalescence matrix: staircase structure.** Matrix M for the 2+2 coalescence pattern: particles 1 and 2 merge into heir H_1 ; particles 3 and 4 merge into heir H_2 . Heir columns (yellow) contain transition probabilities $p(x_i \rightarrow H_j)$. Ghost columns show the staircase pattern: entries with row index $i < \text{rank}(g)$ (blue, solid lines) have $-t^- \cdot p(x_i \rightarrow g)$; entries with $i \geq \text{rank}(g)$ (orange, dashed lines) have $t^+ \cdot p(x_i \rightarrow g)$. The thick staircase line separates the two regions.

1.3.3. *The ghost-free formula.* For applications, one typically wants the probability of heir positions alone, summed over all ghost positions. Integrating out the ghosts yields the *ghost-free coalescence formula* (Section 8): a closed-form determinant whose heir columns contain transition probabilities and whose ghost columns contain cumulative distribution functions in a staircase pattern—alternating between F and $F - 1$.

1.4. **Relation to classical determinantal formulas.** The coalescence formula generalizes the Karlin–McGregor / Lindström–Gessel–Viennot (LGV) determinant. The proof of the Karlin–McGregor theorem [KM59] is a segment-swap argument: configurations where particles cross are paired by exchanging their trajectories at the first crossing time, and the pairs cancel in the signed sum of the determinant. Lindström [Lin73] independently discovered the same cancellation argument for general weighted digraphs in the setting of matroid theory. Gessel and Viennot [GV85]

applied it to lattice path counting, where the cancellation becomes a sign-reversing involution on path tuples; the result is now known as the LGV lemma. Stembridge [Ste90] extended it to D -compatible digraphs; see Krattenthaler [Kra15] for a survey.

In our setting, crossings are not forbidden but prescribed: the ghost configuration specifies which crossings must occur. The involution has the same structure—swap at the first wrong crossing—but “wrong” now means “crossing that violates the prescribed pattern.” The ghost sign ε encodes which crossings are required, and the coefficient extraction selects exactly those terms where particles cross at the right places. When no coalescences occur (no ghosts, every entity an heir), the formula reduces to the classical LGV determinant.

1.5. Prior work. Coalescing Brownian motions were constructed by Arratia [Arr79]. Exact distributional results have since been obtained through several independent lines of work.

1.5.1. Gap distributions and the IPDF method. Doering and ben-Avraham [DA88] introduced the *inter-particle distribution function* (IPDF) method, computing nearest-neighbor gap distributions for diffusion-limited coalescence; ben-Avraham [Avr98] extended this to the full hierarchy of empty-interval probabilities; ben-Avraham and Brunet [AB05] extracted explicit densities for consecutive spacings. See Section 1.7.1 for the ghost-based derivation.

1.5.2. Warren’s determinantal formula. Warren [War07] proved another $n \times n$ determinantal formula for coalescing Brownian motions. His mechanism does not use ghosts: after coalescence, merged particles simply follow the same trajectory. Writing $Z_t(x_i)$ for the position at time t of the survivor containing particle x_i , the joint cumulative distribution function $\Pr(Z_t(x_i) \leq y_i \text{ for all } i)$ can be expressed as a determinant. Since merged particles share a trajectory, the formula does not resolve which particles have coalesced. Assiotis, O’Connell, and Warren [AOW19] extended this via intertwining relations to general diffusions, and Assiotis [Ass18; Ass23] to birth-death chains. The ghost formula provides a finer, pattern-level resolution of Warren’s determinant (see Section 1.7.1).

1.5.3. Pfaffian point processes. A different question arises for infinitely many particles: what is the statistical structure of the surviving positions? Tribe and Zaboronski [TZ11] proved that under the maximal entrance law for coalescing Brownian motions (particles starting from every point of \mathbb{R}), the surviving positions form a Pfaffian point process with an explicit 2×2 matrix kernel involving the complementary error function. Garrod, Poplavskyi, Tribe, and Zaboronski [GPTZ18] extended this to continuous-time random walks on \mathbb{Z} with spatially inhomogeneous rates and arbitrary deterministic initial conditions. Their framework also covers annihilation and mixed coalescence-annihilation and, as shown by Tribe and Zaboronski [TZ26], all entrance laws—but it requires a time-homogeneous Markov generator with a specific algebraic structure (the spin-pair identity). Using the Pfaffian structure, Fomichov [Fom16] found the exact distribution of the number of surviving particles, and Glinyanaya and Fomichov [GF17] proved a central limit theorem with explicit variance.

The ghost formula and the Pfaffian approach have complementary strengths and neither subsumes the other (Section 1.6). See Section 1.7.3 for the ghost-based derivation.

1.5.4. *The Pólya web.* A key motivating example is the *Pólya web*, introduced by the second-named author [Urb25]: coalescing birth-and-death chains on \mathbb{Z} whose steps follow a Pólya urn rule. Each walk has a Beta-distributed limiting direction, and coalescence corresponds to equality of limits. For pairwise coalescence (composition parts at most 2), the second-named author [Urb25] proves that the joint density of these limiting directions is a determinant whose entries alternate between Beta density functions and cumulative distribution functions—a special case of the ghost-free formula (Section 8), which handles arbitrary compositions and general skip-free processes.

The two proofs reach this determinant from complementary directions. Here, the ghost formula—valid for any planar weighted graph—is the starting point: summing out ghost positions converts transition-probability columns into cumulative-sum columns, yielding the ghost-free formula (Section 8). The proof in [Urb25] instead begins with the Karlin–McGregor determinant for non-colliding walks (a pure density determinant), then handles coalescence by conditioning on whether two adjacent walks separate or merge; the resulting integral converts density entries into cumulative distribution entries, one coalescence pair at a time.

Beyond the shared determinantal formula, the two approaches lead to different applications: the companion papers [Śni26a; Śni26c] by the first-named author develop gap distributions and Pfaffian point process structure, while [Urb25] derives the distribution of the number of surviving trees, concentration inequalities, and edge scaling to a Yule web of coalescing branching processes. The companion paper [BSTU26] brings these threads together for the Pólya web.

1.6. Scope and structure of the method. Beyond the specific formulas, the ghost method has several structural features—wide applicability, algebraic flexibility, and exactness—that enable the results of the companion papers.

1.6.1. *Wide scope.* The ghost formula requires exactly the same assumptions as the Karlin–McGregor theorem [KM59]: order preservation, the strong Markov property, and the requirement that meeting times are stopping times. It therefore shares the same broad scope: lattice random walks with arbitrary inhomogeneous transition probabilities, birth-death chains, Brownian motion, and more generally any *skip-free* Markov process (transitions only to neighboring states, so that particles cannot change order without first meeting). The companion papers apply the same formulas, without modification, to symmetric and asymmetric random walks, Brownian motion, Pólya walks [Urb25], and birth-death chains with site-dependent rates. By contrast, the Pfaffian point process approach of Tribe and Zaboronski [TZ11] and Garrod, Poplavskiy, Tribe, and Zaboronski [GPTZ18] requires access to the Markov generator and its spin-pair identity, and the IPDF method [DA88; Avr98] relies on Brownian-motion-specific integrals. Conversely, the analytic approaches reach results that the ghost method does not: mixed coalescence-annihilation and all entrance laws [TZ26]. Neither framework subsumes the other.

1.6.2. *Algebraic structure.* The ghost formula is a determinant, and determinants admit row and column operations. This creates room for algebraic manipulation

that is absent from non-determinantal approaches. When two starting points or two ending points nearly coincide, one can subtract the corresponding rows or columns and divide by the spacing; in the limit, one row becomes a derivative. This confluent-limit technique is how the companion paper [Šni26a] recovers the Tribe–Zaboronski Wronskian kernel [TZ11] from the ghost-free formula (Section 8): subtracting adjacent rows converts pairs of transition probabilities into density-and-derivative pairs, producing the characteristic 2×2 block structure of the Pfaffian kernel. The same mechanism, with forward differences replacing derivatives, yields the discrete Pfaffian point process kernel [Šni26c].

1.6.3. *Exact formulas and the end-game.* Because the ghost formula is exact—not an asymptotic expansion or a scaling limit—one can perform algebraic rearrangements before taking any limits. The companion papers exploit this in three ways. First, [Šni26c] expresses each cumulant of the number of basin boundaries as an exact multidimensional integral over transition densities; rearranging the order of integration converts these into gap-dependent factors, and a combinatorial selection rule guarantees that only finitely many terms contribute, each with a convergent integral. The result is that all cumulants grow at most linearly in the interval length, which gives a central limit theorem by a purely combinatorial route, without generator-based PDE methods. Second, summing the ghost formula over all coalescence patterns and using multilinearity of the determinant collapses the columns one by one, recovering Warren’s determinantal formula [War07] as an algebraic consequence. Third, a deterministic relabeling (cancellative labeling [Šni26c]) converts pairwise coalescence into total annihilation, extracting Pfaffian structure directly from the determinant.

1.6.4. *Conceptual explanations.* Several results in the coalescing particle literature—Warren’s CDF determinant [War07], the Pfaffian point process structure of Tribe and Zaboronski [TZ11] and Garrod, Poplavskiy, Tribe, and Zaboronski [GPTZ18], the IPDF gap distributions [DA88; Avr98], and the central limit theorem of Glinyanaya and Fomichov [GF17]—were first established by analytic or PDE methods. The ghost framework provides alternative, combinatorial proofs of each (see Section 1.7 for details). Beyond reproving known results, the combinatorial route reveals structural connections that are not visible from the analytic perspective. The most striking is that the Pfaffian point process structure belongs naturally to basin boundaries, not to the surviving particles; particles inherit it only through a duality that identifies survivors with boundaries of a dual process [Šni26c]. Similarly, the cumulative distribution entries in Warren’s determinant are not an analytic accident but the ghosts’ contribution: integrating out ghost positions converts transition probabilities into cumulative distributions.

At a structural level, the ghost method is a minimal extension of the Karlin–McGregor theorem: the same sign-reversing involution applied to the same determinantal language, with one new ingredient—the ghost particles that restore the matrix to square form.

1.7. **Companion papers.** This paper is part of a series of five (Figure 3). The left column develops exact combinatorial formulas (this paper for coalescence, a companion for annihilation), while the right column applies them to probability (gap distributions, Pfaffian point processes, and a detailed example). The top row is the coalescence story (ghost particle method \rightarrow *wall-particle system*), the bottom row

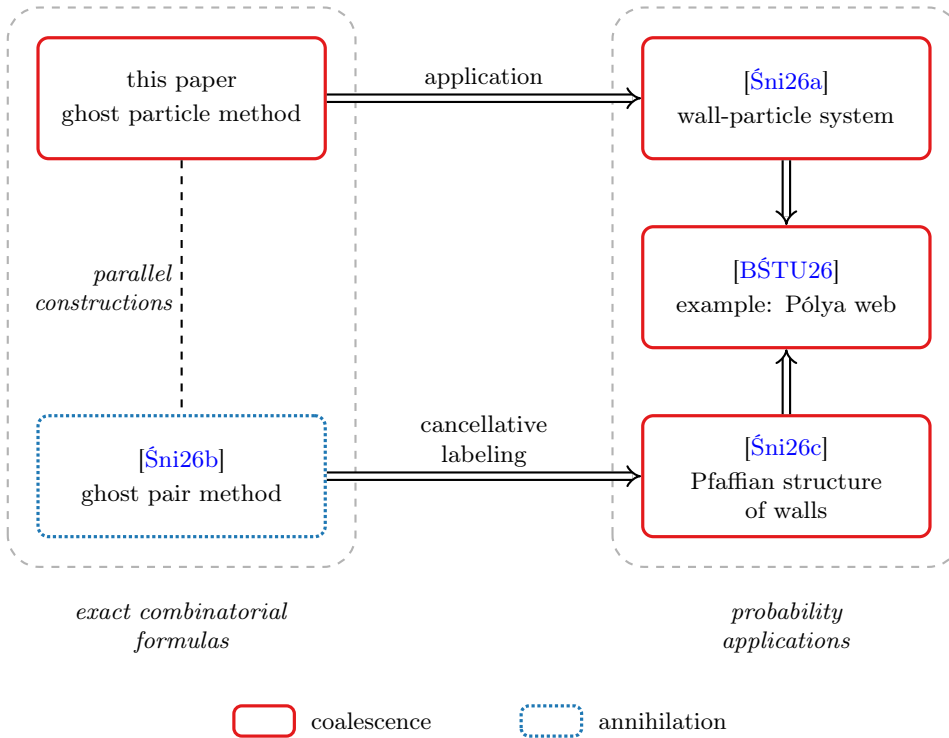


Figure 3. Relations between the companion papers.

starts from annihilation (*ghost pair method*) and uses it to derive Pfaffian structure for the coalescing system. The two rows are independent—neither relies on the other. A fifth paper [BŚTU26] applies both lines of work to the Pólya web.

1.7.1. *Gap distributions and Warren’s formula* [Śni26a]. Under the maximal entrance law (Section 1.5), coalescing particles come down from infinity: at any positive time, only finitely many *survivors* remain per bounded interval. Each survivor attracts a basin of initial particles—those that eventually merge into it. The boundaries between adjacent basins are the *walls* (Figure 4). As particles coalesce, basins merge and walls disappear.

The companion paper [Śni26a] applies the ghost-free formula (Section 8) to this *wall-particle system*—the joint process of survivors and walls. From this it derives the gap distributions and generalizes Warren’s formula (Section 1.5) to arbitrary skip-free processes.

1.7.2. *Annihilation* [Śni26b]. The ghost method also applies to annihilation ($A+A \rightarrow \emptyset$), where both particles are destroyed upon collision. In annihilation, each collision produces a *ghost pair*—two independent random walks starting from the collision point—rather than one heir and one ghost. The companion paper [Śni26b] uses this *ghost pair method* to derive the annihilation formula, which has the same determinantal structure and sign-reversing involution as the coalescence formula. When all $n = 2k$ particles annihilate completely, the determinant reduces to a Pfaffian.

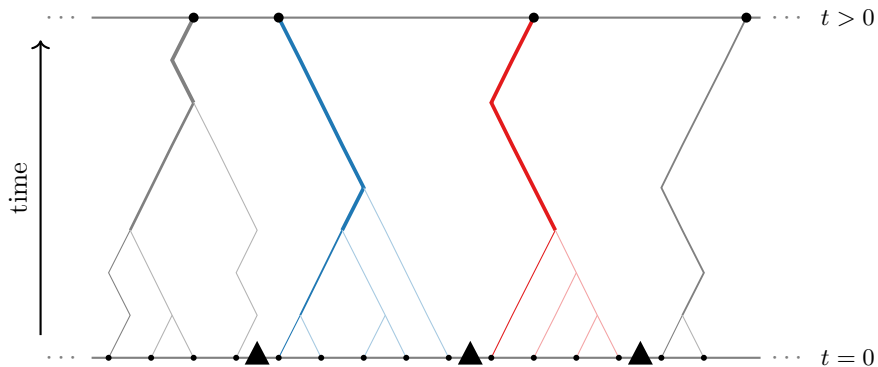


Figure 4. Coalescing random walks starting from every site. Paths merge on contact; line weight increases with each merger. Walls (triangles) mark the boundaries between basins of attraction at $t = 0$; survivors (large dots) are the particles that remain at $t > 0$, one per basin.

Applications include domain wall dynamics in the Ising–Glauber model [GPTZ18], where domain walls annihilate upon meeting.

1.7.3. *Pfaffian structure of walls* [Śni26c]. The companion paper [Śni26c] proves a Pfaffian empty-interval formula for the walls (basin boundaries) of any skip-free coalescing system: the *cancellative labeling* converts pairwise coalescence into total annihilation, and the annihilation formula [Śni26b] supplies the Pfaffian structure. Although the derivation passes through annihilation, the resulting Pfaffian formula describes the *coalescing* system: from it the paper derives explicit cumulant expressions and a central limit theorem for the wall count of the coalescing particles. *Checkerboard duality* then identifies surviving particles with walls of the dual process, transferring the Pfaffian structure to the surviving particle positions. Specializing to Brownian motion under the maximal entrance law recovers the Tribe–Zaboronski kernel [TZ11; GPTZ18].

1.7.4. *The Pólya web* [BŠTU26]. The companion paper [BŠTU26] applies both the coalescence determinant and the Pfaffian machinery to the Pólya web [Urb25], a system of coalescing birth-and-death chains on \mathbb{Z} driven by Pólya urns. The Pólya web is a natural testing ground for the general theory: it is genuinely non-homogeneous (transition probabilities depend on position), yet exactly solvable thanks to the Pólya urn structure; and it mixes the discrete (lattice walks, finitely many survivors at each level) with the continuous (each tree acquires a Beta-distributed asymptotic direction). In suitable projective coordinates, the paper derives exact configuration probabilities, an arcsine law for boundary positions, and Rayleigh spacing for gaps between adjacent opinions. The Pfaffian kernel is built from Beta crossing probabilities, and converges to the complementary error function kernel of coalescing Brownian motions. The number of surviving opinions satisfies a central limit theorem with the universal Fano factor.

1.8. **Companion code.** An accompanying Python package [ŠU26] implements the algorithms of this paper: the attribution map (Definition 4.4), the rehearsal

algorithm (Section 5.3), and the sign-reversing involution (Lemma 5.1). It also verifies the results of the proof—that the involution has the claimed properties (Theorem 5.14 and Lemma 5.1), that the attribution and rehearsal maps are mutual inverses (Proposition 5.15), that the sign identity holds (Proposition 6.1), and that the determinant equals the number of performances (Theorem 3.2)—individually and exhaustively, for up to $n = 5$ particles. Each assertion in the code references the specific proposition or theorem it checks.

Beyond testing, the source code serves as a precise, executable reference: the algorithm implementations and the assertions together provide unambiguous statements of the definitions, the intermediate results, and the main theorem, complementing the prose descriptions in the paper.

1.9. Organization. Section 2 formalizes the coalescence model on weighted directed acyclic graphs, defining *performances* (coalescence histories) and their weights. Section 3 states the coalescence formula. Sections 4 to 6 prove it: Section 4 sets up the combinatorial framework, Section 5 constructs the sign-reversing involution that pairs off-diagonal terms, and Section 6 assembles the proof. Section 7 extends the formula to continuous time and space, covering Brownian motion and birth-death chains. Section 8 integrates out ghost positions, yielding the closed-form ghost-free *coalescence determinant* used in the companion papers.

2. SETUP

The coalescence formula holds for random walks on \mathbb{Z} , Brownian motion on \mathbb{R} , and birth-death chains on arbitrary state spaces. Rather than prove each case separately, we work with *spacetime graphs*—an abstraction that captures two structural properties common to all settings:

- (i) **Planarity:** paths with swapped endpoints must cross, and non-adjacent particles cannot meet without an intermediate particle involved;
- (ii) **Weight-preserving segment swap:** exchanging path segments at crossings preserves total weight.

For discrete models (lattice walks), the spacetime graph is literal: vertices are space-time points, edges are allowed transitions, and edge weights are transition probabilities. For continuous processes (Brownian motion, continuous-time jump processes), the graph is a conceptual tool—the actual proof uses measure-theoretic arguments (Section 7), but the combinatorial structure is identical.

2.1. Spacetime graphs.

Definition 2.1 (Spacetime graph). A *spacetime graph* is a directed acyclic graph $D = (V, E)$ equipped with an edge weight function $w: E \rightarrow R$, where R is a commutative ring (e.g., \mathbb{Z} for combinatorial counting, $\mathbb{R}_{\geq 0}$ for probabilities, or formal power series for generating functions). The directed acyclic graph structure induces a *time ordering*: $u \prec v$ if there is a directed path from u to v . This is a partial order; for the sign-reversing involution (Section 5), we fix a linear extension of \prec . The phrase “first crossing” means first in this linear order; the proof works for any such extension.

Definition 2.2 (Paths and weights). A *path* from x to y is a sequence of vertices $(v_0, v_1, \dots, v_\ell)$ with $v_0 = x$, $v_\ell = y$, and each $(v_i, v_{i+1}) \in E$. The *weight* of a path is

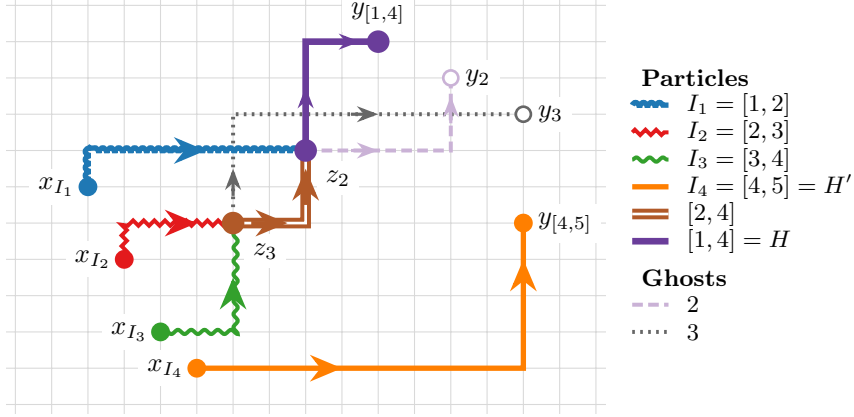


Figure 5. **Running example: coalescence pattern 3+1.** A *performance* records the collision structure on a spacetime graph—here the lattice \mathbb{Z}^2 with North/East steps. Four particles start at x_{I_1} , x_{I_2} , x_{I_3} , x_{I_4} (leaves, colored dots). Particles I_2 and I_3 meet at z_3 , merging into $[2, 4]$; then I_1 and $[2, 4]$ meet at z_2 , forming heir $H = [1, 4]$ which ends at the root $y_{[1,4]}$. Particle I_4 does not coalesce; it is heir $H' = [4, 5]$. Ghost paths (dashed/dotted) emanate from merger points: ghost 2 is born at z_2 , ghost 3 at z_3 . Note that ghost 3 crosses through particle I_1 —ghosts are non-interacting. See Figure 6 for the labeling scheme.

the product of its edge weights:

$$w(P) = \prod_{i=0}^{\ell-1} w(v_i \rightarrow v_{i+1}).$$

The *path generating function* from x to y is

$$W(x \rightarrow y) = \sum_{\substack{P \text{ path} \\ \text{from } x \text{ to } y}} w(P),$$

the sum over all directed paths from x to y .

Example 2.3 (Main examples).

- *Product spacetime* $\mathbb{Z} \times \mathbb{Z}_{\geq 0}$: vertices (x, t) with edges to $(x', t+1)$ for allowed transitions.
- *Checkerboard lattice*: vertices where $x + t$ is even, with edges to $(x \pm 1, t + 1)$ (simple random walk, see Figure 1).

Figure 5 illustrates these concepts with a running example: four particles coalescing on the lattice \mathbb{Z}^2 with North and East steps. We will use this example throughout the paper.

2.2. Planarity. The classical LGV lemma counts *non-intersecting* paths, so it only needs one geometric condition: paths with swapped endpoints must cross. For interacting particles, we need a second condition controlling *which* particles can collide.

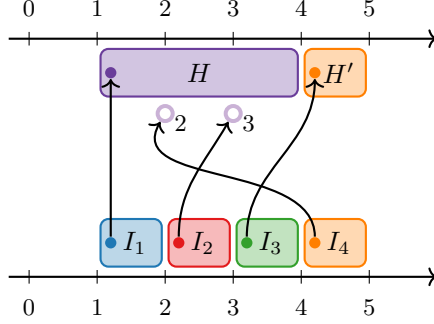


Figure 6. **Interval labeling and the final state.** Initial particles (*actors*) are labeled by unit intervals; final entities (*roles*) by intervals (heirs) or junction points (ghosts). This diagram shows the same example as Figure 5. Here $n = 4$ (Example 2.8): initial particles $\mathcal{A} = \{I_1, I_2, I_3, I_4\}$ and final entities $\mathcal{R} = \{H, 2, 3, H'\}$. Heirs $H = [1, 4]$ and $H' = [4, 5]$; ghosts appear at junctions 2 and 3. The min function is indicated by the small dots: each interval $[a, b]$ has a dot at position a (its left endpoint), so min reads off the horizontal coordinate of the dot. Arrows show one bijection $\pi: I_1 \mapsto H, I_2 \mapsto 3, I_3 \mapsto H', I_4 \mapsto 2$ between the initial particles and the final entities. Under min, this becomes the permutation $1 \mapsto 1, 2 \mapsto 3, 3 \mapsto 4, 4 \mapsto 2$ —the 3-cycle $(2\ 3\ 4)$ with sign $+1$.

Definition 2.4 (Source and target sets). The *source set* $\mathcal{X} \subseteq V$ and *target set* $\mathcal{Y} \subseteq V$ are each equipped with a linear order \prec .

Definition 2.5 (Planar configuration). The pair $(\mathcal{X}, \mathcal{Y})$ is *planar* if:

- (P1) **Crossing property.** For $x \prec x'$ in \mathcal{X} and $y' \prec y$ in \mathcal{Y} (targets swapped), every path from x to y intersects every path from x' to y' .
- (P2) **Consecutive collision property.** For $x \prec x' \prec x''$ in \mathcal{X} , if paths from x and x'' meet at vertex v , then every path from x' must pass through v or intersect one of those paths before v .

The crossing property (P1) is Stembridge’s D -compatibility [Ste90]: when paths have swapped endpoints, they must meet somewhere. Stembridge’s key insight was formulating this condition abstractly for general acyclic digraphs, rather than relying on specific lattice geometry. This abstraction is what enables generalizations to new settings—including ours.

The consecutive collision property (P2) is our addition for interacting particles. It ensures that non-adjacent particles cannot collide without involving intermediate particles—a physical constraint reflecting that particles on a line cannot jump over each other. Classical LGV does not need this because it forbids all crossings; we need it because collisions are allowed but must respect the spatial ordering (see Figure 7).

2.3. Interval labels. Fix source vertices $x_1 \preceq x_2 \preceq \dots \preceq x_n$ in \mathcal{X} . The coalescence structure is tracked using interval labels (see Figure 6).

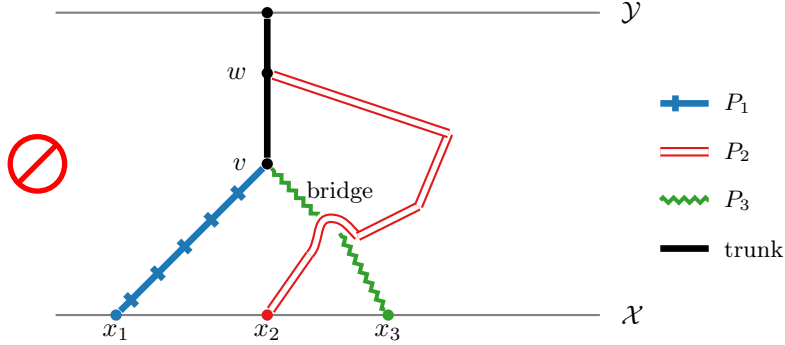


Figure 7. **The consecutive collision property.** A forbidden configuration: paths P_1 and P_3 meet at v and continue as a trunk. Path P_2 attempts to “tunnel” around P_3 without crossing, then hit the trunk at w . Condition (P2) in Definition 2.5 forbids this: P_2 must cross P_1 or P_3 before v .

Definition 2.6 (Interval labeling). The initial particle (also called *actor*) starting at x_j is labeled by the unit interval $I_j = [j, j + 1]$; we also write x_{I_j} or x_I for the starting position of actor I . The set of actors is $\mathcal{A} = \{I_1, \dots, I_n\}$. The *junction points* are $\mathcal{J} = \{2, 3, \dots, n\}$ —the shared endpoints between consecutive intervals.

When particles $[a, b]$ and $[b, c]$ collide, both disappear and two new entities are born: an *heir* labeled $[a, c]$ (the union), and a *ghost* labeled by the junction point b (the dissolved boundary). In particular, particles that share a starting position coalesce instantly: the coalescence occurs at time zero, and the heir and ghost emerge from the shared vertex.

This labeling lets us trace origins: a particle labeled $[a, c]$ arose from the merger of all initial particles I_j with $a \leq j < c$. Similarly, ghost b records which coalescence created it: the one between the particles on either side of junction b .

Intuition: the heir $[a, c]$ is “the owner of the interval $[a, c]$,” and ghost b is “attached to the boundary that used to separate $[a, b]$ from $[b, c]$.”

2.3.1. *Label order.* The *label order* \triangleleft (read “label-less-than”) orders intervals and junctions by position: $[a, b] \triangleleft g$ iff $b \leq g$, and $g \triangleleft [a, b]$ iff $g \leq a$. We write \trianglelefteq for the reflexive closure (allowing $f \trianglelefteq f$), and $\triangleright, \trianglerighteq$ for the reversed relations. For example, $I_1 \triangleleft 2 \triangleleft I_2 \triangleleft 3 \triangleleft I_3$.

The function \min identifies intervals and junctions with $[n] = \{1, \dots, n\}$: junction $k \mapsto k$, interval $[a, b] \mapsto a$. Under this map, coalescence of $[a, b]$ with $[b, c]$ into $[a, c]$ (ghost b) becomes: a persists, b transitions from particle label to ghost label.

2.4. **Final state.** The *final state* \mathcal{F} specifies:

- The *ghost set* $\mathcal{G} \subseteq \mathcal{J}$: junctions where coalescence occurred;
- The *heir set* \mathcal{H} : maximal intervals $[a, b] \subseteq [1, n + 1]$ not containing any junction outside \mathcal{G} ;
- The *role set* $\mathcal{R} = \mathcal{H} \cup \mathcal{G}$: the final entities;
- Final positions y_f for each entity $f \in \mathcal{R}$.

We assume heir positions respect label order: if $H \triangleleft H'$, then $y_H \prec y_{H'}$. This is not a restriction: by the planarity condition (Definition 2.5), any physically realizable coalescence pattern must satisfy this ordering. The cardinality is $|\mathcal{R}| = |\mathcal{H}| + |\mathcal{G}| = k + m = n$.

Definition 2.7 (Heir function). For ghost $g \in \mathcal{G}$, we write $\text{heir}(g)$ for the unique heir interval containing junction g : $\text{heir}(g) = [a, b]$ where $a < g < b$.

Example 2.8 (Coalescence pattern 3+1). Consider $n = 4$ initial particles (Figures 5 and 6):

$$\mathcal{A} = \{I_1, I_2, I_3, I_4\}.$$

Suppose I_1 and I_2 coalesce (ghost at junction 2), then the merged particle $[1, 3]$ coalesces with I_3 (ghost at junction 3), while I_4 remains separate. The final state has:

$$\mathcal{G} = \{2, 3\}, \quad \mathcal{H} = \{[1, 4], [4, 5]\}, \quad \mathcal{R} = \{[1, 4], 2, 3, [4, 5]\}.$$

Two heirs and two ghosts. Under the identification with $[4]$:

- $\mathcal{A} = \{I_1, I_2, I_3, I_4\} \longleftrightarrow \{1, 2, 3, 4\}$ (particle I_j maps to j);
- $\mathcal{R} = \{[1, 4], 2, 3, [4, 5]\} \longleftrightarrow \{1, 2, 3, 4\}$ (heir $[1, 4] \mapsto 1$, heir $[4, 5] \mapsto 4$, ghosts map to themselves).

Any bijection $\pi: \mathcal{A} \rightarrow \mathcal{R}$ is thus a permutation of $\{1, 2, 3, 4\}$.

The ghost set \mathcal{G} determines a composition $c_1 + \dots + c_k = n$, where c_j counts the initial particles that merge into the j th heir. In the example above, $\mathcal{G} = \{2, 3\}$ gives composition 3+1; Figure 2 illustrates the 2+2 pattern.

Definition 2.9 (Rank). The *rank* $\text{rank}(f)$ of entity $f \in \mathcal{R}$ is its numerical label under the min identification: $\text{rank}(g) = g$ for ghosts, $\text{rank}([a, b]) = a$ for heirs.

2.5. Performances. The classic Lindström–Gessel–Viennot lemma abstracts non-colliding random walks into a graph-theoretic problem: counting tuples of vertex-disjoint paths. We seek an analogous abstraction for coalescing particles. The dynamics—particles moving, meeting, merging—is replaced by a static combinatorial object: a forest of genealogy trees recording which particles merged, together with ghost paths recording where each ghost traveled (Figure 5). This abstraction is the *performance*.

Definition 2.10 (Genealogy tree). A *genealogy* for an heir $H \in \mathcal{H}$ records which initial particles merged to form H . It is an oriented tree T embedded in D with:

- **Leaves:** starting positions x_I for particles I that merge into H ;
- **Internal vertices:** merger points where two or more particles merge and continue as one;
- **Root:** the final position y_H of the heir;
- **Edges:** directed paths in D —particle trajectories.

Each vertex v of T carries a *label* I_v : the union of the initial intervals of all particles that have merged by v . When two initial particles share a starting vertex, that vertex serves as both leaf and internal vertex; the incoming paths have length zero.

Proposition 2.11 (Consecutivity). *Under the planarity assumption (Definition 2.5), the label I_v at each vertex is an interval: if particles A and C have both reached v , then so has every particle B between them.*

Proof. If $A \triangleleft C$ both reach v , then any intermediate B must pass through v or intersect one of their paths before v (by condition (P2) from Definition 2.5; see Figure 7). In the first case, B reaches v directly. In the second case, B coalesces with A or C before v and continues as part of the merged particle, which reaches v . In either case, B has reached v . \square

Definition 2.12 (Genealogy forest). A *genealogy forest* is a collection $\mathcal{T} = \{T_H : H \in \mathcal{H}\}$ of genealogies satisfying:

- **Partition:** Every initial particle belongs to exactly one tree;
- **Non-intersection:** Trees are vertex-disjoint.

Definition 2.13 (Ghost paths). For each ghost $g \in \mathcal{G}$, let c_g be the internal vertex where junction g was dissolved. The *ghost path* Γ_g is a directed path in D from c_g to y_g . Ghost paths are non-interacting: they may pass through any vertices freely.

Definition 2.14 (Performance). A *performance* \mathcal{P} consists of:

- a genealogy forest \mathcal{T} ;
- a ghost path Γ_g for each ghost $g \in \mathcal{G}$.

The *weight* of the performance is:

$$w(\mathcal{P}) = \prod_{\text{tree edges } e} w(e) \cdot \prod_{g \in \mathcal{G}} w(\Gamma_g).$$

2.6. Ghost sign.

Definition 2.15 (Ghost sign). The *ghost sign* $\varepsilon: \mathcal{G} \rightarrow \{+1, -1\}$ is:

$$\varepsilon(g) = \begin{cases} +1 & \text{if } y_g \preceq y_{\text{heir}(g)} \text{ (ghost left of heir),} \\ -1 & \text{if } y_{\text{heir}(g)} \prec y_g \text{ (ghost right of heir).} \end{cases}$$

Two distinct orderings appear in our setup (Figure 6). Actors and final entities carry *labels*—intervals and junctions ordered by \triangleleft . But entities also have *spatial positions*—vertices in the spacetime graph ordered by \prec .

In the classical LGV lemma, these orderings coincide: the permutation π has the same meaning whether interpreted as permuting labels or as crossing paths.

With ghosts, this alignment breaks: heirs are sorted by spatial position, but a ghost might end up to the left or the right of its heir, regardless of their labels. The ghost sign ε captures precisely this discrepancy.

3. THE COALESCENCE FORMULA

The introduction used transition probabilities $p(x \rightarrow y)$ for random walks. In the general graph setting, we use the path generating function

$$W(x \rightarrow y) = \sum_{P: x \rightarrow y} w(P),$$

which sums edge weights over all paths from x to y . For random walks, $W = p$.

3.1. Formal variables. For each ghost $g \in \mathcal{G}$, introduce formal parameters t_g^+ and t_g^- . These track which side of the heir each ghost ends up on.

3.2. **The matrix.** Define the $n \times n$ matrix M with:

- Rows indexed by particles $I \in \mathcal{A}$ (in label order);
- Columns indexed by final entities $f \in \mathcal{R}$ (in rank order: each heir followed by its attached ghosts).

Entries:

$$M_{I,f} = \begin{cases} W(x_I \rightarrow y_H) & \text{if } f = H \text{ is an heir,} \\ t_g^+ \cdot W(x_I \rightarrow y_g) & \text{if } f = g \text{ is a ghost and } I \trianglerighteq g, \\ -t_g^- \cdot W(x_I \rightarrow y_g) & \text{if } f = g \text{ is a ghost and } I \triangleleft g. \end{cases}$$

Each ghost column g has a “staircase” pattern: the dividing line falls at the row where I transitions from $I \triangleleft g$ to $I \trianglerighteq g$. Above the staircase: entries have $-t_g^-$. Below: entries have t_g^+ (Figure 2).

Remark 3.1 (Equivalence of notations). Under the min identification (Section 2.3), the label order condition $I \trianglerighteq g$ is equivalent to $\min(I) \geq g = \text{rank}(g)$. This recovers the rank-based formulation from the introduction (Section 1.3).

3.3. **The theorem.**

Theorem 3.2 (Coalescence formula with ghosts). *The generating function for coalescence performances is:*

$$Z = \left[\prod_{g \in \mathcal{G}} t_g^{\varepsilon(g)} \right] \det(M).$$

This generalizes Theorem 1.1. The coefficient extraction $[\prod_g t_g^{\varepsilon(g)}]$ selects terms matching the ghost sign vector (Section 1.3.2).

The proof occupies Sections 4 to 6: Section 4 sets up the correspondence between performances and castings, Section 5 develops the sign-reversing involution that cancels failed castings, and Section 6 completes the argument. An outline appears at the start of Section 4.

4. PROOF SETUP

4.1. **Performances vs. castings.** Throughout the proof, we fix a final state \mathcal{F} (Section 2.4): the ghost set \mathcal{G} , the heir set \mathcal{H} , and final positions y_f for each $f \in \mathcal{R}$. The generating function $Z = Z_{\mathcal{F}}$ counts *performances* for \mathcal{F} : complete descriptions of what coalescences occur and where the final entities end up.

4.1.1. *Performances.* A performance is **role-based**—it specifies outcomes without tracking which particle “becomes” which final entity. In probabilistic terms, each performance is an elementary event in the sample space.

Consider the coalescence pattern 2+1 from Figure 1: three particles start at $x_1 < x_2 < x_3$, particles 1 and 2 coalesce, and the system ends with two heirs at $y_1 < y_2$ and a ghost at z . The role-based description records:

- A coalescence occurred at spacetime point c .
- The heir from that coalescence reached y_1 ; the ghost reached z .
- The remaining particle reached y_2 .
- The paths taken by each entity.

This description says nothing about *which* of particles 1 and 2 became the heir and which became the ghost—only that the coalescence produced one of each. This discontinuity is reflected in Figure 1: the line styles entering the coalescence differ from those leaving it, emphasizing that identities do not persist across the event.

This role-based viewpoint is natural for stating results: we ask “*what is the probability that coalescences occur at these locations and entities end at these positions?*” without specifying identities. In theater terms: the script specifies where each character should stand at the final scene, but does not specify which actor plays which character.

4.1.2. *The determinant perspective.* The determinant $\det(M)$ gives something different: a signed sum over *castings*—assignments of particles to final positions via non-interacting paths. A casting is **actor-based**: it tracks each particle’s complete trajectory. In Figure 8, line styles persist through the coalescence, revealing who went where.

The proof establishes a bijection between performances and a subset of castings (the “successful” ones—those producing valid coalescence patterns; see Definition 5.7), while the remaining “failed” castings cancel in pairs (Figure 9).

4.1.3. *Outline of the argument.* The argument has four parts:

- (1) **Castings from the determinant.** The Leibniz expansion produces castings; coefficient extraction selects *candidate* castings consistent with the ghost configuration.
- (2) **Attribution.** Every performance determines a casting by tracking which particle ends where. *Attribution* always produces a candidate casting.
- (3) **Rehearsal.** We attempt to interpret each casting as a performance by scanning crossings in temporal order. This may succeed (“successful casting”) or fail (“failed casting”).
- (4) **The sign-reversing involution.** Failed castings pair up via segment swap and cancel. Only successful castings remain.

A unifying observation ties these together: both attribution and rehearsal are *online algorithms* that process crossings chronologically, one at a time, with no memory beyond their current state. The bijection between performances and successful castings, and the involution on failed castings, both follow from *local invertibility*: at each crossing, the two algorithms perform the same operation, so interleaving them step by step gives the identity.

4.2. Castings: what the determinant gives.

4.2.1. *The Leibniz expansion.* The Leibniz formula expands the determinant as a sum over bijections $\pi: \mathcal{A} \rightarrow \mathcal{R}$:

$$\det(M) = \sum_{\pi} \operatorname{sgn}(\pi) \prod_{I \in \mathcal{A}} M_{I, \pi(I)}.$$

Each matrix entry $M_{I, f}$ is either a path generating function $W(x_I \rightarrow y_f)$ (for heir columns) or $\pm t_g^{\pm} \cdot W(x_I \rightarrow y_g)$ (for ghost columns). Expanding:

$$\det(M) = \sum_{\pi} \operatorname{sgn}(\pi) \sum_{\mathbf{P}} (\text{sign factors}) \cdot w(\mathbf{P}),$$

where $\mathbf{P} = \{P_I\}_{I \in \mathcal{A}}$ is a family of paths with P_I running from x_I to $y_{\pi(I)}$.

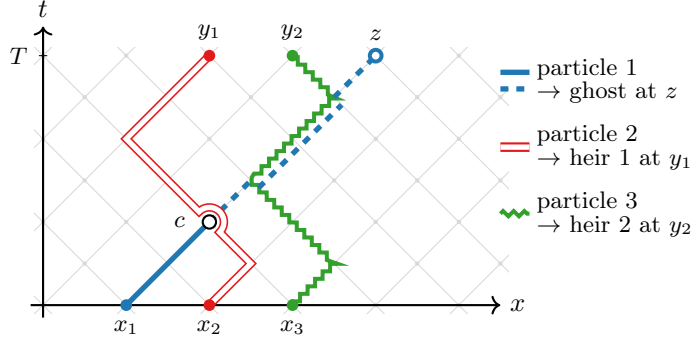


Figure 8. **Successful casting: actor-based view of a performance.** Same collision as Figure 1, but now tracking identities. The bijection π assigns: particle 1 (solid) \rightarrow ghost at z ; particle 2 (double) \rightarrow heir at y_1 ; particle 3 (zigzag) \rightarrow heir at y_2 . Line styles persist through the collision, revealing who went where. The arc at c shows particle 2 wrapping around to continue as heir; particle 1 continues as the ghost.

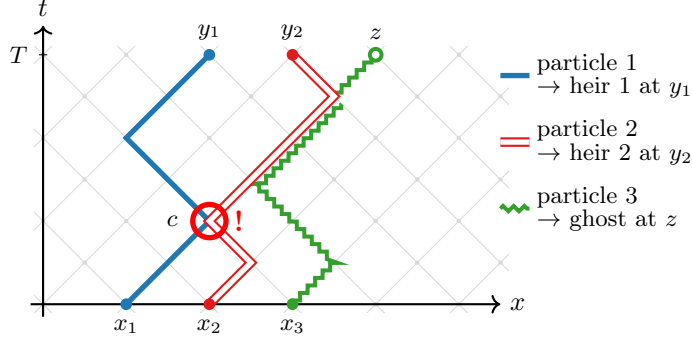


Figure 9. **Failed casting: a spurious crossing.** Same endpoints as Figure 8, but different paths and bijection. Here π assigns particle 1 $\rightarrow y_1$, particle 2 $\rightarrow y_2$, particle 3 $\rightarrow z$. At the crossing c , particles 1 and 2 meet, but neither is destined for the ghost position z —both go to heir positions. This crossing is *spurious*: it cannot be interpreted as a collision, so rehearsal fails. Failed castings cancel in pairs via segment swap (Section 5.1).

Definition 4.1 (Casting). A *casting* (π, \mathbf{P}) consists of:

- A bijection $\pi: \mathcal{A} \rightarrow \mathcal{R}$ from actors (initial particles) to roles (final entities);
- A path family $\mathbf{P} = \{P_I\}_{I \in \mathcal{A}}$ where P_I goes from x_I to $y_{\pi(I)}$.

The *weight* is $w(\mathbf{P}) = \prod_I w(P_I)$, the product of path weights. Under the min identification (Section 2.3, Figure 6), π becomes a permutation of $[n]$; the *sign* $\text{sgn}(\pi)$ is the usual permutation sign.

Crucially, castings are *pure geometry*: the paths are non-interacting. They may cross freely—a crossing is simply a point where two paths share a vertex, with no

physical consequence. There is no coalescence physics yet: the determinant gives us n non-interacting walkers, not an interacting particle system.

4.2.2. *Candidate bijections.* The coefficient extraction $[\prod_g t_g^{\varepsilon(g)}]$ selects specific bijections. Each ghost column g contributes t_g^+ in rows where $I \supseteq g$ and $-t_g^-$ where $I \triangleleft g$. Extracting $t_g^{\varepsilon(g)}$ selects bijections where the *performer* $\pi^{-1}(g)$ (the initial particle that π assigns to ghost role g) satisfies:

- If $\varepsilon(g) = +1$: $\pi^{-1}(g) \supseteq g$ (performer right of ghost junction);
- If $\varepsilon(g) = -1$: $\pi^{-1}(g) \triangleleft g$ (performer left of ghost junction).

Intuitively, the candidacy condition ensures that required crossings exist: a particle destined for a ghost position must have a path that crosses another path from the correct direction.

Definition 4.2 (Source sign). For a bijection π and ghost $g \in \mathcal{G}$, the *source sign* is:

$$\delta_\pi(g) = \begin{cases} +1 & \text{if } g \triangleleft \pi^{-1}(g) \text{ (performer right of ghost),} \\ -1 & \text{if } \pi^{-1}(g) \triangleleft g \text{ (performer left of ghost).} \end{cases}$$

The ghost sign $\varepsilon(g)$ compares the ghost's final *spatial position* to its heir's *spatial position*. The source sign $\delta_\pi(g)$ compares the ghost's junction *label* to its performer's interval *label*.

Definition 4.3 (Candidate bijection). A bijection $\pi: \mathcal{A} \rightarrow \mathcal{R}$ is a *candidate* if $\varepsilon(g) = \delta_\pi(g)$ for all ghosts $g \in \mathcal{G}$. We write $\Pi_{\mathcal{F}}$ for the set of candidate bijections.

The coefficient extraction $[\prod_g t_g^{\varepsilon(g)}]$ selects exactly the candidate bijections. Each ghost column g contributes an independent formal variable (t_g^+ or t_g^- depending on the row), so the extraction acts column by column: for each ghost g , it retains the rows where the formal variable matches $t_g^{\varepsilon(g)}$, which is precisely the candidacy condition $\varepsilon(g) = \delta_\pi(g)$.

4.2.3. *The restricted Leibniz expansion.* Crucially, the candidacy condition depends only on π , not on the paths \mathbf{P} . This gives the determinant expansion a clean structure:

$$(4.1) \quad \sum_{\substack{(\pi, \mathbf{P}) \\ \pi \in \Pi_{\mathcal{F}}}} \text{sgn}(\pi) w(\mathbf{P}) = \sum_{\pi \in \Pi_{\mathcal{F}}} \text{sgn}(\pi) \prod_{I \in \mathcal{A}} W(x_I \rightarrow y_{\pi(I)}).$$

The product is over non-interacting paths—a product of path generating functions. The coefficient extraction selects exactly the candidate bijections, yielding a “restricted Leibniz expansion”: like a determinant, but summing only over $\Pi_{\mathcal{F}}$ rather than all of S_n .

4.3. **Attribution: performance to casting.** A performance specifies coalescence locations and final positions but not identities. *Attribution* constructs the underlying casting by tracking which particle ends where. We first describe what happens at a single coalescence of two particles, then explain how to combine these local operations.

4.3.1. *The two-particle case.* By consecutivity (Proposition 2.11), coalescing particles are always adjacent intervals. Write I^- and I^+ for the left and right incoming intervals at a coalescence, H for the heir, and g for the ghost. (In interval notation: $I^- = [a, g]$, $I^+ = [g, c]$, $H = [a, c]$.)

Four path segments meet at the coalescence vertex (Figures 10a and 11a): two incoming and two outgoing (one to the ghost position y_g , one continuing as the heir toward further coalescences or the final position). The *swap principle* connects incoming to outgoing segments by exchanging sides: the particle from I^- (the lower-labeled interval) continues toward the farther final position; the particle from I^+ continues toward the nearer position.

The ghost sign $\varepsilon(g)$ encodes whether the ghost ends left or right of the heir, determining the gluing (Figures 10b and 11b):

- $\varepsilon(g) = +1$ (ghost left of heir): the right interval I^+ becomes the ghost;
- $\varepsilon(g) = -1$ (ghost right of heir): the left interval I^- becomes the ghost.

This defines a *local bijection* from $\{I^-, I^+\}$ to $\{H, g\}$ (Figures 10c and 11c).

4.3.2. *From local to global.*

Definition 4.4 (Attribution). *Attribution* constructs a casting (π, \mathbf{P}) from a performance by applying the local gluing rule at each coalescence vertex.

For each initial particle I , follow its path through all coalescence vertices, applying the local gluing rule at each one. The path terminates either at an heir position or at a ghost position. This process produces:

- the endpoint $\pi(I)$ (the final entity where I ends up);
- the glued path P_I from x_I to $y_{\pi(I)}$.

Iterating over all initial particles yields the output of attribution: a casting $(\pi, (P_I)_{I \in \mathcal{A}})$.

High-indegree vertices. By consecutivity, when $m > 2$ intervals meet simultaneously they form a consecutive sequence I_1, \dots, I_m . Process them in label order with nested pairings: $((I_1 I_2) I_3) \cdots I_m$. At each step, two intervals enter: the leftmost unpaired interval and the heir from the previous pairing (or, at the first step, the two leftmost intervals). The corresponding ghost (identified by its junction label) exits, and the heir continues to pair with the next interval. After $m - 1$ steps, $m - 1$ ghosts have been created and one final heir remains.

For example, if three intervals I_1, I_2, I_3 meet at a single vertex: first I_1 and I_2 are paired, producing a ghost at their shared junction and a merged interval; then the merged interval pairs with I_3 , producing a second ghost and the final heir. This left-to-right order matches the lexicographic tie-breaking used in rehearsal (Section 5.3), ensuring that attribution and rehearsal agree.

4.3.3. *Examples.*

Example 4.5 (Single coalescence: Figure 1). In Figure 1, particles I_1 and I_2 coalesce at c . The ghost ends at $z > y_1$, so $\varepsilon(g) = -1$ (ghost right of heir). By the swap principle, the left particle I_1 becomes the ghost. The output is the casting $(\pi, (P_{I_1}, P_{I_2}, P_{I_3}))$ where:

- P_{I_1} : from x_1 to c , then to z ; endpoint $\pi(I_1) = g$;
- P_{I_2} : from x_2 to c , then to y_1 ; endpoint $\pi(I_2) = H_1$;
- P_{I_3} : from x_3 to y_2 (no coalescence); endpoint $\pi(I_3) = H_2$.

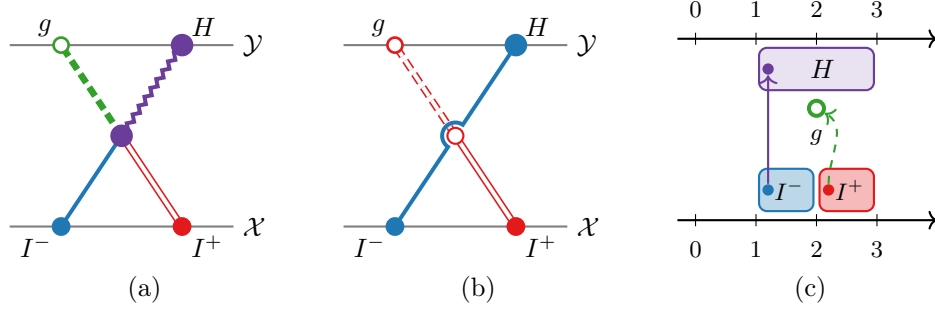


Figure 10. **Two-particle coalescence, case $\varepsilon(g) = +1$: ghost left of heir.** (a) Schema: four distinct styles indicate entities do not persist past coalescence; the dashed line shows the ghost path. (b) Attribution via path gluing: the swap principle routes the particle from I^- to y_H (solid), while the particle from I^+ continues as a ghost to y_g (dashed). Following paths determines the bijection. (c) Bijection π_+ : $I^- \mapsto H, I^+ \mapsto g$.

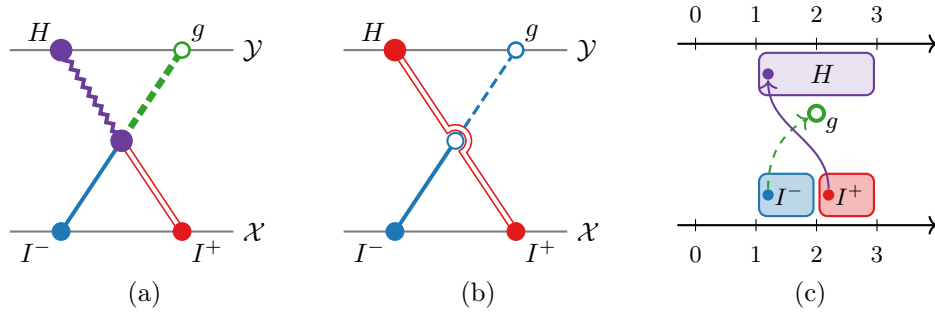


Figure 11. **Two-particle coalescence, case $\varepsilon(g) = -1$: ghost right of heir.** (a) Schema: four distinct styles indicate entities do not persist past coalescence; the dashed line shows the ghost path. (b) Attribution via path gluing: the swap principle routes the particle from I^+ to y_H (double line), while the particle from I^- continues as a ghost to y_g (dashed). Following paths determines the bijection. (c) Bijection π_- : $I^- \mapsto g, I^+ \mapsto H$.

See Figure 8.

Example 4.6 (Full coalescence: Figures 12 and 13). Three particles I_1, I_2, I_3 all coalesce into one heir $H = [1, 4]$, with ghost signs $\varepsilon(2) = +1$ (ghost 2 left of heir) and $\varepsilon(3) = -1$ (ghost 3 right of heir).

In Figure 12, junction 3 fires first: I_2 and I_3 meet. Since $\varepsilon(3) = -1$, the left particle I_2 becomes the ghost. Next, junction 2 fires: I_1 meets the merged interval (carried by I_3). Since $\varepsilon(2) = +1$, the right particle (I_3) becomes the ghost. The output is the casting $(\pi_1, (P_{I_1}, P_{I_2}, P_{I_3}))$:

- P_{I_1} : from x_1 to junction 2, then to y_H ; endpoint $\pi_1(I_1) = H$;

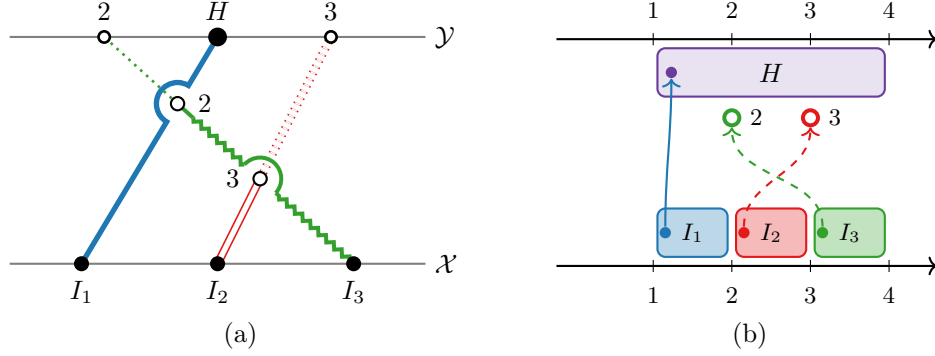


Figure 12. **Full coalescence, bijection π_1 : junction 3 fires first.** Three particles coalesce into one heir $H = [1, 4]$, creating ghosts at junctions 2 and 3. (a) Attribution: following each path through all collisions determines the bijection. Junction 3 fires first (I_2 and I_3 meet), then junction 2 (I_1 meets the merged $[2, 4]$). (b) Bijection π_1 : $I_1 \mapsto H$, $I_2 \mapsto 3$, $I_3 \mapsto 2$. Line styles: I_1 solid, I_2 double, I_3 zigzag; dotted = ghost.

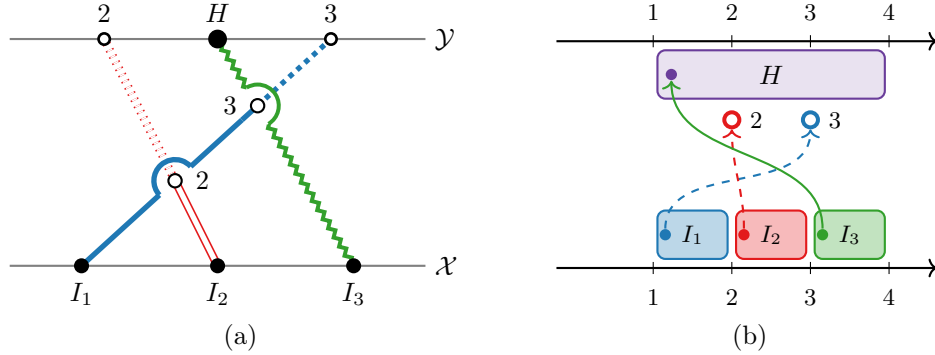


Figure 13. **Full coalescence, bijection π_2 : junction 2 fires first.** Same final state as Figure 12, different collision order. (a) Attribution: junction 2 fires first (I_1 and I_2 meet), then junction 3 (I_3 joins). Following paths yields a different bijection. (b) Bijection π_2 : $I_1 \mapsto 3$, $I_2 \mapsto 2$, $I_3 \mapsto H$. Both π_1 and π_2 are successful castings for the same final state.

- P_{I_2} : from x_2 to junction 3, then to y_3 ; endpoint $\pi_1(I_2) = 3$;
- P_{I_3} : from x_3 to junction 3, to junction 2, then to y_2 ; endpoint $\pi_1(I_3) = 2$.

In Figure 13, the same final state arises from a different coalescence order (junction 2 fires first), producing a different casting $(\pi_2, (P'_{I_1}, P'_{I_2}, P'_{I_3}))$ with $\pi_2(I_1) = 3$, $\pi_2(I_2) = 2$, $\pi_2(I_3) = H$.

4.3.4. Candidacy emerges.

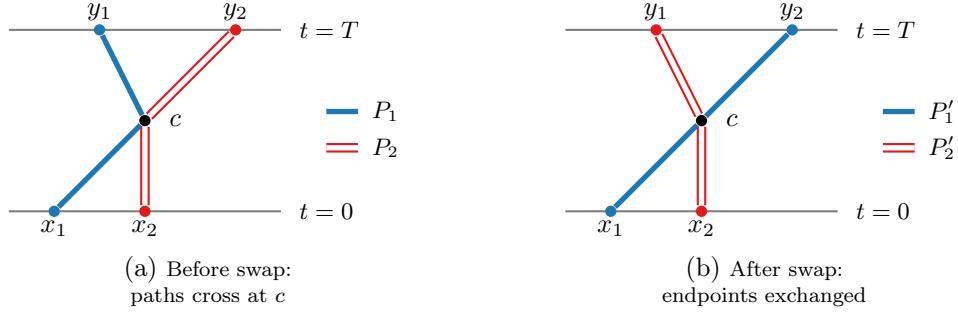


Figure 14. **Segment swap: the sign-reversing involution.**

(a) Paths P_1 (solid) and P_2 (double) cross at vertex c . (b) After the swap, final segments are exchanged: P'_1 follows P_1 to c , then P_2 's tail to y_2 ; P'_2 follows P_2 to c , then P_1 's tail to y_1 . The bijection updates to $\pi' = (1\ 2) \circ \pi$, reversing the sign. Failed castings cancel in pairs via this operation.

Proposition 4.7. *Let $\mathcal{C} = (\pi, \mathbf{P})$ be the casting from attributing a performance. Then π is a candidate.*

Proof. Consider a coalescence creating ghost g , with adjacent intervals I^- and I^+ as above.

When $y_{\text{heir}(g)} \prec y_g$ (ghost ends right, so $\varepsilon(g) = -1$), the left particle I^- terminates at g . Tracing I^- back through any earlier coalescences, we reach some initial particle I with $I \subseteq I^-$, so in particular $I \triangleleft g$. Since $\pi(I) = g$, we have $\delta_\pi(g) = -1$, matching $\varepsilon(g)$.

When $y_g \preceq y_{\text{heir}(g)}$ (ghost ends left, so $\varepsilon(g) = +1$), the right particle I^+ terminates, and tracing back gives an initial particle I with $I \triangleright g$, so $\delta_\pi(g) = +1$, again matching. \square

5. THE SIGN-REVERSING INVOLUTION

We now develop the sign-reversing involution that cancels failed castings.

The algorithm tracks actor ownership through the **Active** dictionary; when starting positions are distinct, the actor owning each path can also be read from its starting vertex (Section 5.5).

5.1. Segment swap. *Segment swap* is a transformation on castings. Given a casting (π, \mathbf{P}) and two actors I, J whose paths cross, segment swap exchanges their suffixes at the first crossing v : the new path P'_I follows P_I to v , then continues along P_J ; symmetrically for P'_J . The bijection updates accordingly: $\pi' = (I\ J) \circ \pi$, exchanging the destinations of I and J . See Figure 14.

Lemma 5.1 (Segment swap is involutive and sign-reversing). *Segment swap is:*

- (i) **An involution:** swapping twice recovers the original.
- (ii) **Weight-preserving:** same path segments, same total weight.
- (iii) **Sign-reversing:** $\text{sgn}(\pi') = -\text{sgn}(\pi)$.

Proof. (i) Swapping at v twice restores the original paths and bijection. (ii) The path segments are redistributed but the multiset of edges is unchanged, so total weight is preserved. (iii) The updated bijection $\pi' = (I J) \circ \pi$ differs from π by a transposition. \square

5.2. Local involutions.

5.2.1. *Overview.* Following Lindström [Lin73] and Gessel–Viennot [GV85], we construct a sign-reversing involution on candidate castings. The building block is a *local involution* $\iota_{I,J}$ defined for any pair of actors I, J : it applies segment swap at the first crossing of their paths, unless the paths do not cross or the swap breaks candidacy. Local involutions are defined on any casting—they do not depend on the algorithm state. The algorithm (Section 5.3) composes them into a global involution by processing crossings chronologically.

5.2.2. *Definition.* For actors $I \neq J$, the *local involution* $\iota_{I,J}$ is:

$$(5.1) \quad \iota_{I,J}(\mathcal{C}) = \begin{cases} \text{segment swap of } I, J & \text{if } P_I \text{ and } P_J \text{ cross and} \\ & \text{the swapped casting is a candidate,} \\ \mathcal{C} & \text{otherwise.} \end{cases}$$

5.2.3. *Fixed points and valid crossings.* When two adjacent actors at junction g have crossing paths, we say that the crossing is *valid* if one of the actors is destined for the ghost role at g —that is, the crossing corresponds to a genuine coalescence event. The key lemma says that valid crossings are precisely the fixed points of the local involution: segment swap at a valid crossing would break candidacy (so ι does nothing), while swap at a spurious crossing preserves candidacy (so ι pairs the casting with another).

Lemma 5.2 (Swap criterion at a crossing). *Let $\mathcal{C} = (\pi, \mathbf{P})$ be a candidate casting, and let I and J be adjacent actors at junction g , whose paths cross. Then $\iota_{I,J}$ fixes the casting if and only if $g \in \{\pi(I), \pi(J)\}$.*

Proof. The swap exchanges the roles of I and J : $\pi'(I) = \pi(J)$ and $\pi'(J) = \pi(I)$.

Key observation. For any ghost $h \neq g$, the source sign $\delta_\pi(h)$ is unchanged by the swap. Since I and J are adjacent at g , they lie on the same side of every other junction $h \neq g$: $I \triangleleft h \iff J \triangleleft h$. So even if the performer changes from I to J or vice versa, the source sign is preserved.

Case $g \in \{\pi(I), \pi(J)\}$. Say $\pi(J) = g$. After swap: $\pi'(I) = g$. But $I \triangleleft g$ while $J \triangleright g$, so the source sign $\delta_\pi(g)$ changes. Since candidacy requires $\varepsilon(g) = \delta_\pi(g)$, the swapped casting violates candidacy. Hence \mathcal{C} is a fixed point.

Case $g \notin \{\pi(I), \pi(J)\}$. The performer $\pi^{-1}(g)$ is unchanged by the swap. By the key observation, source signs for all ghosts are preserved. The swapped casting remains a candidate, so the swap is performed. \square

Remark 5.3. When actors I and J are non-adjacent (representing merged intervals in the algorithm), the key observation needs an additional assumption: that the performers of ghosts inside $I^- \cup I^+$ are distinct from I and J . This *performer invariant* is stated and verified in the algorithm section (Lemma 5.5).

5.3. The algorithm. *Rehearsal* attempts to interpret a candidate casting as a performance by processing crossings in temporal order. The same algorithm defines the global involution ι (Theorem 5.14): successful castings are fixed points, while failed castings pair up via segment swap.

5.3.1. *High-level view.* The algorithm processes crossings chronologically. At each step, the first crossing involves two adjacent *active intervals* $I^- \triangleleft I^+$ (intervals that have not yet been consumed by a coalescence) at junction g .

Definition 5.4 (Valid and spurious crossings). A crossing between adjacent active intervals I^- and I^+ at junction g is *valid* if one of them is destined for the ghost role at g , i.e., $g \in \{\sigma(I^-), \sigma(I^+)\}$, where σ is the reduced bijection defined below. Otherwise the crossing is *spurious*.

A valid crossing becomes a coalescence: one path exits as a ghost path, the other represents the merged interval, and the system shrinks. A spurious crossing triggers segment swap: the algorithm returns the swapped casting and terminates (a *failed* casting). The algorithm terminates *successfully* when no crossings remain.

5.3.2. *The procedure.*

Input: A candidate casting (π, \mathbf{P}) , where $\pi: \mathcal{A} \rightarrow \mathcal{R}$ is a candidate bijection (fixed throughout).

State variables. The casting (π, \mathbf{P}) describes the full system; the algorithm maintains a *reduced* version tracking the active (not yet consumed) intervals:

- **Active:** a dictionary mapping *active intervals* (possibly merged) to their paths—the reduced path family (keys evolve as intervals merge, values remain fixed);
- σ : a dictionary mapping active intervals to roles—the reduced bijection (initially $\sigma = \pi$; keys evolve as intervals merge, values remain fixed). At each step, $\sigma(\tilde{I}) = \pi(I)$ where I is the *representative actor* of \tilde{I} : the initial particle whose path $\text{Active}(\tilde{I})$ currently represents \tilde{I} ;
- \mathcal{P} : the performance under construction (coalescence events and ghost paths recorded so far).

Since starting positions are distinct, the actor owning each path in **Active** is determined by the starting vertex.

Algorithm.

(1) Initialize:

$$\text{Active} \leftarrow \{I_j \mapsto P_{I_j}\}_{j=1}^n, \quad \sigma \leftarrow \pi, \quad \mathcal{P} \leftarrow \emptyset.$$

(2) While paths in **Active** cross:

- (a) Find the first crossing among active paths: the adjacent pair $I^- \triangleleft I^+$ at junction g whose paths $P = \text{Active}(I^-)$ and $Q = \text{Active}(I^+)$ cross at the earliest vertex v . If multiple pairs share the same vertex, select the lexicographically minimal pair (I^-, I^+) ; this ensures adjacency (Section 5.3.5). Let I, J be the actors owning P, Q (determined by starting vertices).
- (b) **Test:** Is $g \in \{\sigma(I^-), \sigma(I^+)\}$? Equivalently (Corollary 5.6): is \mathcal{C} a fixed point of $\iota_{I,J}$?

- If **no** (spurious crossing): apply $\iota_{I,J}$ and return the swapped casting. The algorithm terminates (*failure*).
- If **yes** (valid crossing)—record a coalescence:
 - Let **ghost** be the interval among $\{I^-, I^+\}$ with $\sigma(\text{ghost}) = g$, and let **survivor** be the other.
 - *Ghost path*: Delete **ghost** from **Active**; the suffix of its path from v becomes the ghost path Γ_g .
 - *Genealogy*: Record v as an internal vertex (merger point) of the genealogy tree; the segments of P and Q from their most recent coalescence vertex (or starting point) to v become tree edges.
 - Form the merged interval $H \leftarrow I^- \cup I^+$. Update **Active**: set $\text{Active}(H) \leftarrow \text{Active}(\text{survivor})$; delete the entry for **survivor**.
 - Update σ : set $\sigma(H) \leftarrow \sigma(\text{survivor})$; delete entries for **ghost** and **survivor**. Return to step 2.
- (3) **Success** (no active paths cross)—the remaining active paths (suffixes from the last coalescence to the heir endpoints) become the final edges of the genealogy trees. Return (π, \mathbf{P}) (a fixed point) together with the performance \mathcal{P} . The algorithm terminates (*success*).

5.3.3. *The performer invariant.* At each algorithm step, the actors I and J representing I^- and I^+ may not be adjacent initial particles—they can be separated by junctions of earlier coalescences. As noted in Remark 5.3, the key lemma then needs an additional assumption: performers of ghosts inside $I^- \cup I^+$ are distinct from I and J . The algorithm maintains this as an invariant.

Lemma 5.5 (Interior ghost performers are distinct from current actors). *At each iteration of the algorithm, let I^- and I^+ be the adjacent active intervals at junction g , represented by actors I and J . For every ghost $h \neq g$ inside $I^- \cup I^+$: the performer $\pi^{-1}(h)$ is distinct from I and J .*

Proof. A ghost h inside I^- was the junction between two intervals that merged in an earlier iteration. At that step, the two representatives were some initial particles I' and J' , and the crossing at h was valid (otherwise the algorithm would have failed before reaching junction g). So $\pi^{-1}(h) \in \{I', J'\}$: it is the representative that was consumed as the ghost performer, and no longer represents any active interval. Since I and J represent current active intervals, $\pi^{-1}(h) \notin \{I, J\}$. \square

Corollary 5.6 (Swap criterion for non-adjacent actors). *At each iteration of the algorithm, let I^- and I^+ be adjacent active intervals at junction g , represented by actors I and J whose paths cross. Then $\iota_{I,J}$ fixes the casting if and only if $g \in \{\sigma(I^-), \sigma(I^+)\}$.*

Proof. The proof of the key lemma (Lemma 5.2) extends verbatim: for ghosts h outside $I^- \cup I^+$, both I and J lie on the same side of h (since $I \subseteq I^-$ and $J \subseteq I^+$); for ghosts h inside $I^- \cup I^+$, the performer invariant (Lemma 5.5) gives $\pi^{-1}(h) \notin \{I, J\}$, so the swap does not affect h . \square

5.3.4. *Successful castings.*

Definition 5.7 (Successful casting). A candidate casting is *successful* if the algorithm processes all crossings as valid coalescences without encountering a spurious one. Equivalently, a successful casting is a fixed point of ι .

Remark 5.8 (Composing local involutions). In the classical Lindström–Gessel–Viennot proof, a single segment swap at the first crossing suffices: every crossing is spurious, so the involution is a one-step operation. Our involution generalizes this by composing local involutions: valid crossings are consumed (the system shrinks via coalescence); only the first spurious crossing triggers a segment swap. The idea of composing involutions to achieve global cancellation has antecedents in the involution principle of Garsia and Milne [GM81]; the specific chronological-processing mechanism used here is new.

5.3.5. *Why the first crossing is between adjacent intervals.* By consecutivity (Proposition 2.11), paths of non-adjacent intervals cannot cross before paths between them do. If intervals A and C are non-adjacent with B between them, then before A and C can cross, either A must cross B or C must cross B . This holds for the initial configuration and is preserved after each merger (the reduced system inherits consecutivity).

5.3.6. *Crossings must exist.*

Lemma 5.9 (Crossings exist). *If some active interval I satisfies $\sigma(I) = g$ for a ghost role g , then at least two active paths must cross.*

This ensures the algorithm cannot “stall” with unfilled ghost roles. The proof is by contradiction: if no paths cross, then left-to-right order forces every actor to map to an heir, leaving no actor for the ghost.

Proof. We prove the lemma for the initial step (all actors active, $\sigma = \pi$). The same argument applies at any later step: after k coalescences, the reduced system has active intervals $\tilde{I}_1 \triangleleft \cdots \triangleleft \tilde{I}_{n-k}$ (possibly merged) with reduced bijection σ mapping them to the remaining roles. The reduced bijection is a candidate for the reduced final state (Proposition 4.7 applied to the reduced system). The proof below uses only the candidacy of the bijection, the label ordering of active intervals, and the heir position assumption—all of which hold for the reduced system.

Suppose no paths cross. Then paths preserve left-to-right order: $I \triangleleft J$ implies $y_{\pi(I)} \prec y_{\pi(J)}$. This single observation constrains the bijection π in two ways:

- (i) *Heir ordering:* $\pi^{-1}(H_1) \triangleleft \cdots \triangleleft \pi^{-1}(H_\ell)$. Otherwise left-to-right order would contradict the heir position assumption (Section 2.4).
- (ii) *Ghost constraint.* For ghost g with heir $H = \text{heir}(g)$, left-to-right order gives

$$\pi^{-1}(g) \triangleleft \pi^{-1}(H) \iff y_g \prec y_H \iff \varepsilon(g) = +1.$$

Candidacy ($\varepsilon(g) = \delta_\pi(g)$) gives $\varepsilon(g) = +1 \iff g \triangleleft \pi^{-1}(g)$. Combining, exactly one of the following holds:

- (G1) $g \triangleleft \pi^{-1}(g) \triangleleft \pi^{-1}(H)$, or
- (G2) $\pi^{-1}(H) \triangleleft \pi^{-1}(g) \triangleleft g$.

Claim. *Under (i) and (ii), for each $k \in \{0, 1, \dots, n\}$: $\ell \geq k$ and $\pi(I_1) = H_1, \dots, \pi(I_k) = H_k$.*

The case $k = n$ gives the contradiction: every actor maps to a heir, leaving no actor for ghost g . The quantification over all k is needed only to carry the induction.

Proof of Claim. By induction on k .

Base case ($k = 0$). Trivial.

Inductive step. Assume the claim holds for $k - 1$. We show $\ell \geq k$ and $\pi(I_k) = H_k$.

Step 1: Existing heirs have no ghosts. Suppose heir H_j with $j \leq k-1$ has a ghost; take j minimal. By induction, $\pi^{-1}(H_j) = I_j$. Let $g' = \min \text{ghosts}(H_j)$. By minimality of j , heirs H_1, \dots, H_{j-1} are trivial (single-particle intervals), so H_j starts at position j . The ghost constraint (ii) for g' requires:

- (G1) $\pi^{-1}(g') \triangleleft I_j$: impossible, since I_1, \dots, I_{j-1} all map to heirs by induction.
- (G2) $I_j \triangleleft \pi^{-1}(g') \triangleleft g'$: write $H_j = [j, b]$ (since H_j starts at j), so $g' = j+1$. But this requires $\pi^{-1}(g') = I_m$ with $m > j$ and $m+1 \leq g' = j+1$, giving $m \leq j$. Contradiction.

Hence $H_i = [i, i+1]$ for $i \leq k-1$: existing heirs are trivial.

Step 2: $\pi(I_k)$ is not a ghost. Suppose $\pi(I_k) = g$ for some ghost g with $\text{heir}(g) = H_j$. Since existing heirs have no ghosts, $j \geq k$. Thus H_j starts at position $\geq k$, so $g \geq k+1$. The ghost constraint (ii) requires:

- (G1) $g \triangleleft I_k$: impossible, since $g \geq k+1$ and $I_k = [k, k+1]$ give $I_k \triangleleft g$.
- (G2) $\pi^{-1}(H_j) \triangleleft I_k$: then we need $\pi^{-1}(H_j) \in \{I_1, \dots, I_{k-1}\}$, but these map to H_1, \dots, H_{k-1} by induction, not to H_j with $j \geq k$.

Both fail, so $\pi(I_k)$ is an heir.

Step 3: Identification of $\pi(I_k)$. By Step 2, $\pi(I_k) = H_m$ for some m . By induction, H_1, \dots, H_{k-1} are already assigned to I_1, \dots, I_{k-1} , so $m \geq k$; in particular, $\ell \geq k$ (heir H_k exists). If $m > k$, heir ordering (i) gives $\pi^{-1}(H_k) \triangleleft \pi^{-1}(H_m) = I_k$, forcing $\pi^{-1}(H_k) \in \{I_1, \dots, I_{k-1}\}$ —but these map to H_1, \dots, H_{k-1} . Hence $m = k$: $\pi(I_k) = H_k$. This completes the inductive step, and hence the Claim. \square

This completes the proof of Lemma 5.9. \square

5.3.7. Successful castings yield valid performances.

Proposition 5.10. *Let \mathcal{C} be a successful candidate casting. The performance produced by the algorithm has the prescribed final state \mathcal{F} .*

Proof. Lemma 5.9 guarantees that all ghost assignments are processed: if any interval were still destined for a ghost position, crossings would remain among active paths.

Ghost set. Each valid crossing occurs at a junction $g \in \{\sigma(I^-), \sigma(I^+)\}$, so only junctions in \mathcal{G} are processed.

Ghost endpoints. At the coalescence creating ghost g , the actor $\pi^{-1}(g)$ is still active. The ghost path Γ_g is the suffix of $P_{\pi^{-1}(g)}$ from the crossing vertex, ending at y_g .

Genealogy trees. Coalescences occur at exactly the junctions in \mathcal{G} . After all coalescences, the active intervals are the maximal intervals not containing any junction outside \mathcal{G} —exactly \mathcal{H} .

Heir endpoints. For heir $H \in \mathcal{H}$, the initial particle I with $\pi(I) = H$ is never consumed as a ghost, so its path reaches y_H . \square

5.3.8. *The online perspective.* Both attribution and rehearsal are *online algorithms*: they process crossings chronologically, one at a time, and their state between consecutive crossings is the same triple (Active, σ , \mathcal{P}) that the algorithm maintains. We call this triple the *intermediate state*. It encodes a “centaur”: the bottom part \mathcal{P} is the performance constructed so far; the top part (Active, σ) is the remaining casting data not yet consumed.

Attribution as a fold. Given a performance, attribution processes coalescence vertices in temporal order. At each coalescence between adjacent intervals I^- and I^+ at junction g : the ghost sign $\varepsilon(g)$ determines which interval becomes the ghost (Section 4.3), the ghost path is recorded in \mathcal{P} , and the active intervals and reduced bijection update. Between consecutive coalescences, the active paths are non-crossing segments of the performance's genealogy trees.

Rehearsal as a fold. Given a candidate casting, rehearsal processes the first crossing among active paths. At each valid crossing between I^- and I^+ at junction g : the reduced bijection σ determines which interval becomes the ghost, the ghost path is recorded in \mathcal{P} , and the active intervals update. Between consecutive valid crossings, the remaining active paths carry no earlier crossings (since the algorithm always selects the first).

Local invertibility. At a single crossing vertex, the attribution step and the rehearsal step perform the same operation: both use the local bijection from $\{I^-, I^+\}$ to $\{H, g\}$ determined by $\varepsilon(g)$. Attribution reads this from the performance (which side the ghost exits); rehearsal reads it from σ (which interval is destined for the ghost role). For a successful casting arising from a performance, these agree (Proposition 4.7): candidacy ensures that the ghost assignment from σ matches the ghost sign $\varepsilon(g)$.

Interleaving principle. To verify that the composition of two online algorithms is the identity, it suffices to interleave them step by step: apply the first algorithm to one crossing, immediately feed the output to the second algorithm, and verify that the original input is recovered. Since neither algorithm has memory beyond the intermediate state, and since each local step is invertible, the global round-trip follows by induction on the number of crossings.

5.3.9. Restricted commutativity.

Remark 5.11. The involution theorem (Theorem 5.14) is proved using the online perspective of Section 5.3.8, which avoids explicit commutativity arguments. For completeness, we record restricted commutativity as an independent verification that the algorithm's validity decisions are robust under segment swap.

The local involutions commute whenever the algorithm might need to apply both.

Definition 5.12 (Restricted commutativity). The local involutions satisfy *restricted commutativity* if: whenever the algorithm (applied to casting \mathcal{C}) checks both pairs (I, J) and (K, L) , then

$$(\iota_{I,J} \circ \iota_{K,L})(\mathcal{C}) = (\iota_{K,L} \circ \iota_{I,J})(\mathcal{C}).$$

Lemma 5.13 (Restricted commutativity). *The local involutions $\{\iota_{I,J}\}$ for coalescence satisfy restricted commutativity.*

Proof. Suppose the algorithm, when processing a casting $\mathcal{C} = (\pi, \mathbf{P})$, checks pair (I, J) first and later checks pair (K, L) . We must show that

$$\iota_{I,J}(\iota_{K,L}(\mathcal{C})) = \iota_{K,L}(\mathcal{C}),$$

i.e., if \mathcal{C} is a fixed point of $\iota_{I,J}$, then so is $\iota_{K,L}(\mathcal{C})$.

Let $\mathcal{C}' = (\pi', \mathbf{P}') = \iota_{K,L}(\mathcal{C})$. If $\iota_{K,L}$ fixes \mathcal{C} (no swap), then $\mathcal{C}' = \mathcal{C}$ and the claim holds trivially. Otherwise, $\iota_{K,L}$ swaps P_K and P_L at their first crossing.

Paths before the swap point. The swap only exchanges segments of P_K and P_L after their first crossing. Since (I, J) was checked before (K, L) , the first crossing of P_I and P_J occurs before the first crossing of P_K and P_L . Thus \mathbf{P} and \mathbf{P}' are identical in the region where P_I and P_J first meet—the swap does not affect this geometry.

Fixed-point status at (I, J) . By Corollary 5.6, the fixed-point condition for $\iota_{I,J}$ is equivalent to validity of the crossing: $g \in \{\sigma(I^-), \sigma(I^+)\}$ where g is the junction. Since $\sigma(I^-) = \pi(I)$ and $\sigma(I^+) = \pi(J)$ (the algorithm maintains this correspondence), we equivalently need $g \in \{\pi(I), \pi(J)\}$.

Consider how the swap $\iota_{K,L}$ affects π :

- If $\{I, J\} \cap \{K, L\} = \emptyset$: the values $\pi(I)$ and $\pi(J)$ are unchanged, so $g \in \{\pi(I), \pi(J)\} \iff g \in \{\pi'(I), \pi'(J)\}$.
- If $|\{I, J\} \cap \{K, L\}| = 1$: say $J = K$ (other cases are symmetric). Since \mathcal{C} is a fixed point at (I, J) , the crossing is valid, so $g \in \{\pi(I), \pi(J)\}$. If $\pi(J) = g$, then J is the ghost and is removed from Active, so it cannot appear in a later pair—contradiction. If $\pi(I) = g$, then I is removed but J stays active; the overlap can occur. The swap $\iota_{J,L}$ exchanges $\pi(J) \leftrightarrow \pi(L)$ but leaves $\pi(I) = g$ unchanged, so the fixed-point condition $g \in \{\pi(I), \pi(J)\}$ is preserved.
- If $\{I, J\} = \{K, L\}$: impossible. After a valid crossing at (I, J) , one actor deactivates and the pair cannot reappear.

In all cases, the validity condition is preserved.

Performer invariant for \mathcal{C}' . The argument above applied Corollary 5.6 to the swapped casting \mathcal{C}' , but the corollary's proof relies on the performer invariant (Lemma 5.5), which was established for the original casting \mathcal{C} . One might hope to invoke Lemma 5.5 directly for \mathcal{C}' by re-running the algorithm on \mathcal{C}' from scratch. This does not work: the algorithm's validity tests depend on π , and since π' differs from π at K and L , a crossing involving K or L before step (I, J) could change from valid to spurious (or vice versa), causing the algorithm's execution on \mathcal{C}' to diverge entirely from its execution on \mathcal{C} .

Instead, we transfer the invariant from \mathcal{C} to \mathcal{C}' directly. The swap changes π only at K and L : so $\pi'^{-1}(h) \neq \pi^{-1}(h)$ only if $\pi^{-1}(h) \in \{K, L\}$. But by Lemma 5.5, $\pi^{-1}(h)$ was consumed as a ghost performer at an earlier step and is no longer active. Since K and L are active at the later pair (K, L) , we have $\pi^{-1}(h) \notin \{K, L\}$. Hence $\pi'^{-1}(h) = \pi^{-1}(h)$, and the performer invariant is preserved. \square

Theorem 5.14 (Successful castings are the fixed points of ι). *The map ι is a weight-preserving, sign-reversing involution on candidate castings. Its fixed points are exactly the successful castings.*

Proof. Well-defined. The local involution only swaps when the resulting casting is a candidate (Corollary 5.6). The global involution applies at most one local swap, so the output is a candidate casting.

Fixed points. By definition, global fixed points are castings where every crossing is valid—exactly the successful castings (Definition 5.7).

Involution. Let \mathcal{C} have first spurious crossing at pair (I, J) at vertex v , and let $\mathcal{C}' = \iota_{I,J}(\mathcal{C})$. We must show that \mathcal{C}' also has first spurious crossing at (I, J) , so that swapping again recovers the original.

The online perspective (Section 5.3.8) gives a direct argument. Run the algorithm on \mathcal{C} : it processes valid crossings chronologically, building the intermediate state, until it reaches the spurious crossing at v . Now run the algorithm on \mathcal{C}' . The segment swap $\iota_{I,J}$ only modifies path suffixes after v ; all path segments before v are identical in \mathcal{C} and \mathcal{C}' . Therefore the algorithm on \mathcal{C}' encounters the same crossings in the same temporal order before v . At each such crossing, the validity test $g \in \{\sigma(I^-), \sigma(I^+)\}$ gives the same result: the crossing location and the active paths are identical, so the same intervals meet at the same junction, and the reduced bijection σ evolves identically (each valid crossing removes the same ghost and merges the same intervals). The intermediate state at v is therefore the same.

At v , the paths of I and J still cross (the crossing occurs before the swap point), and the crossing is still spurious: swapping exchanges the representatives of I^- and I^+ , so $\sigma(I^-)$ and $\sigma(I^+)$ swap values, but the validity test checks whether $g \in \{\sigma(I^-), \sigma(I^+)\}$, and this set is unchanged. Thus (I, J) is the first spurious crossing for \mathcal{C}' , and $\iota^2 = \text{id}$.

Properties. Weight-preservation and sign-reversal follow from Lemma 5.1. \square

5.4. The performance–casting bijection.

Proposition 5.15. *For fixed final state \mathcal{F} , attribution and successful castings are in bijection: attribution maps performances to successful castings, and the algorithm of Section 5.3 maps successful castings back to performances.*

Proof. Both attribution and rehearsal are online algorithms (Section 5.3.8): they process crossings chronologically, maintaining the same intermediate state $(\text{Active}, \sigma, \mathcal{P})$. We verify both compositions by interleaving the two algorithms step by step.

Rehearsal \circ Attribution = id. Let \mathcal{P}_0 be a performance with coalescences at junctions g_1, \dots, g_m in temporal order. Attribution produces a candidate casting $(\pi, \mathbf{P}) = \mathfrak{A}(\mathcal{P}_0)$ (Proposition 4.7). We show that rehearsal, applied to this casting, recovers \mathcal{P}_0 .

Instead of running attribution to completion and then rehearsal, we interleave them. Before the first coalescence, the intermediate state is the initial one: all intervals active, $\sigma = \pi$, $\mathcal{P} = \emptyset$. We maintain an inductive invariant: just before the k th coalescence, the intermediate state is the same whether reached by attribution (processing the first $k - 1$ coalescences of \mathcal{P}_0) or by rehearsal (processing the first $k - 1$ valid crossings of the casting).

Inductive step. At the k th coalescence, adjacent intervals I^- and I^+ meet at junction g_k . Attribution assigns the ghost role using $\varepsilon(g_k)$: the local bijection $\{I^-, I^+\} \rightarrow \{H, g_k\}$ is determined by the ghost sign. By the inductive hypothesis, rehearsal has the same intermediate state at this moment. The crossing at g_k is the first crossing among active paths: between coalescences g_{k-1} and g_k , the active paths follow edges of distinct genealogy trees or distinct branches of the same tree; these are non-crossing by the vertex-disjointness of trees and the planarity of the embedding. The crossing is valid because $g_k \in \{\sigma(I^-), \sigma(I^+)\}$: candidacy gives $\varepsilon(g_k) = \delta_\pi(g_k)$, which determines which of $\sigma(I^-)$, $\sigma(I^+)$ equals g_k . Rehearsal therefore uses the same local bijection $\{I^-, I^+\} \rightarrow \{H, g_k\}$, producing the same ghost path and the same updated state.

After processing all m coalescences, no active paths cross (the remaining paths are the non-crossing heir suffixes), so rehearsal terminates successfully with performance \mathcal{P}_0 .

Attribution \circ *Rehearsal* = id. Let $\mathcal{C} = (\pi, \mathbf{P})$ be a successful casting, and let \mathcal{P}_0 be the performance produced by rehearsal. By Proposition 5.10, \mathcal{P}_0 has final state \mathcal{F} . We show $\mathfrak{A}(\mathcal{P}_0) = \mathcal{C}$ by the same interleaving.

At each valid crossing between I^- and I^+ at junction g , rehearsal creates a coalescence in \mathcal{P}_0 using σ : the interval with $\sigma(\cdot) = g$ becomes the ghost. Attribution, processing the same coalescence in \mathcal{P}_0 , assigns the ghost role using $\varepsilon(g)$. These agree: candidacy ($\varepsilon(g) = \delta_\pi(g)$) ensures that the ghost sign matches the source sign, and the performer invariant (Lemma 5.5) ensures the performer of g arrives via the correct interval. Specifically, if $\varepsilon(g) = +1$ then $\delta_\pi(g) = +1$, so the performer $\pi^{-1}(g) \triangleright g$ arrives via I^+ ; hence $\sigma(I^+) = g$ —matching attribution’s assignment.

Path recovery. At each coalescence, the ghost path in \mathcal{P}_0 is a suffix of $P_{\pi^{-1}(g)}$ from the crossing vertex. Attribution glues path segments at the same vertices with the same assignments, recovering \mathbf{P} .

Candidacy is preserved. After each step, the reduced system remains a candidate casting: ghost g and its performer are removed, I^- and I^+ merge, and the merged interval inherits the non-ghost role. For any remaining ghost $g' \neq g$, the performer $\pi^{-1}(g')$ and its label ordering relative to g' are unchanged. \square

5.5. Non-distinct starting positions. The distinct-start assumption made at the beginning of this section is not used in the proof: the algorithm tracks ownership through the Active dictionary, not through starting vertices. The remark that “the actor owning each path is determined by the starting vertex” is a geometric observation, not a proof step. When starting positions coincide, the dictionary continues to map each active interval to its path, and the entire argument applies without modification.

6. PROOF OF THE COALESCENCE FORMULA

With the involution established, we complete the proof.

6.1. The sign identity.

Proposition 6.1 (Sign identity). *For any successful casting \mathcal{C} :*

$$\text{sgn}(\pi) = \prod_{g \in \mathcal{G}} \varepsilon(g).$$

Proof. Identify labels with $[n]$ via left endpoints: interval $[a, b]$ maps to a , and junction point g maps to g . Under this identification, $\pi: \mathcal{A} \rightarrow \mathcal{R}$ becomes a permutation of $[n]$.

We prove by induction on $|\mathcal{G}|$. For zero ghosts, $\pi = \text{id}$ and $\text{sgn}(\pi) = +1 = \prod_{g \in \emptyset} \varepsilon(g)$.

For the inductive step, pick a terminal coalescence: one whose heir $H = [a, c]$ is final. At this coalescence, intervals $I^- = [a, g]$ and $I^+ = [g, c]$ merge, creating ghost g . Define a modified final state \mathcal{F}_0 by “undoing” this coalescence: replace H by two heirs I^- and I^+ , and remove ghost g . The bijection π induces a bijection π_0 for this modified state: π_0 agrees with π on all actors except that I^- and I^+ now map to themselves (as heirs in \mathcal{F}_0) rather than to the ghost and heir roles they had in \mathcal{F} . The reduced bijection π_0 is a candidate for \mathcal{F}_0 : for any ghost $g' \neq g$, the performer $\pi_0^{-1}(g') = \pi^{-1}(g')$ and its label ordering relative to g' are unchanged, so $\varepsilon(g') = \delta_{\pi_0}(g')$.

The key observation: under the label-order identification of \mathcal{R} with $\{1, \dots, n\}$, ghost g occupies the position between I^- and I^+ . So:

- If $\varepsilon(g) = -1$: I^- becomes the ghost ($\pi(I^-) = g$), while I^+ maps to heir H . Under identification, $\pi(a) = g$ and $\pi(g) = a$. Thus $\pi = (a\ g) \circ \pi_0$, giving $\text{sgn}(\pi) = -\text{sgn}(\pi_0) = \varepsilon(g) \cdot \text{sgn}(\pi_0)$.
- If $\varepsilon(g) = +1$: I^+ becomes the ghost ($\pi(I^+) = g$), while I^- maps to heir H . Under identification, $\pi(a) = a$ and $\pi(g) = g$; for all other indices, π and π_0 agree (same actors, same roles). Thus $\pi = \pi_0$ as permutations of $[n]$, giving $\text{sgn}(\pi) = \text{sgn}(\pi_0) = \varepsilon(g) \cdot \text{sgn}(\pi_0)$.

By induction, $\text{sgn}(\pi) = \prod_g \varepsilon(g)$. \square

The sign identity implies that all successful castings contribute the same sign—namely $\prod_g \varepsilon(g)$ —to the generating function.

6.2. Completing the proof.

Theorem 6.2 (Performances equal a signed sum over candidates). *The generating function for performances equals*

$$Z = \text{sgn } \mathcal{F} \cdot \sum_{\pi \in \Pi_{\mathcal{F}}} \text{sgn}(\pi) \prod_{I \in \mathcal{A}} W(x_I \rightarrow y_{\pi(I)}),$$

where $\text{sgn } \mathcal{F} = \prod_{g \in \mathcal{G}} \varepsilon(g)$.

Proof. The involution ι (Theorem 5.14) partitions candidate castings into:

- *Fixed points*: successful castings (contribute to sum);
- *Pairs*: failed castings, grouped by segment swap.

Each pair has equal weight and opposite sign, so they cancel. Only successful castings remain.

By the sign identity (Proposition 6.1), every successful casting has $\text{sgn}(\pi) = \prod_g \varepsilon(g) = \text{sgn } \mathcal{F}$. Therefore

$$\begin{aligned} \text{sgn } \mathcal{F} \cdot \sum_{\text{successful}} \text{sgn}(\pi) \cdot w(\mathbf{P}) &= \text{sgn } \mathcal{F} \cdot \text{sgn } \mathcal{F} \cdot \sum_{\text{successful}} w(\mathbf{P}) \\ &= \sum_{\text{successful}} w(\mathbf{P}). \end{aligned}$$

The bijection between successful castings and performances (Proposition 5.15) completes the proof. \square

Proof of Theorem 3.2. By the performance–casting identity (Theorem 6.2), Z equals a signed sum over candidate bijections. The coefficient extraction $[\prod_g t_g^{\varepsilon(g)}]$ selects exactly the candidate bijections (Definition 4.3).

It remains to verify that the sign factors align. In the Leibniz expansion, each ghost column g contributes a factor of -1 for rows above the staircase ($I \triangleleft g$). For a candidate π , the number of such sign flips equals the number of ghosts g with $\pi^{-1}(g) \triangleleft g$, i.e., with $\delta_{\pi}(g) = -1$. Since candidacy gives $\varepsilon(g) = \delta_{\pi}(g)$, the product of these matrix signs is $\prod_{g: \varepsilon(g) = -1} (-1)$. Combined with the permutation sign $\text{sgn}(\pi) = \prod_g \varepsilon(g)$ (Proposition 6.1) and the overall prefactor $\text{sgn } \mathcal{F} = \prod_g \varepsilon(g)$, the total sign is $+1$ for every successful casting.

Therefore $Z = [\prod_g t_g^{\varepsilon(g)}] \det(M)$ equals the sum of path weights over all performances. \square

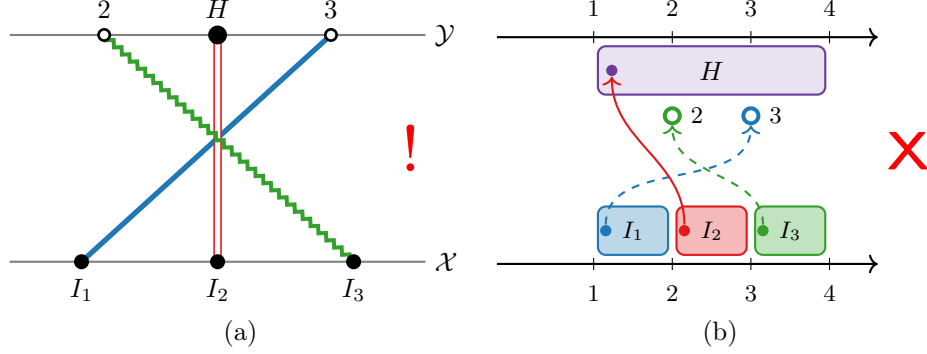


Figure 15. **An unsuccessful candidate: π_3 admits no successful casting.** The bijection $\pi_3: I_1 \mapsto 3, I_2 \mapsto H, I_3 \mapsto 2$ satisfies the candidacy condition but admits no successful casting. (a) Any path family has a spurious crossing—the middle particle I_2 must collide before reaching H . (b) Bijection π_3 . Compare with Figures 12 and 13.

6.3. **Example: full coalescence of three particles.** We return to Example 4.6: three particles I_1, I_2, I_3 coalesce into one heir $H = [1, 4]$, with $\varepsilon(2) = +1$ and $\varepsilon(3) = -1$. The candidacy condition requires:

- $\pi^{-1}(2) \supseteq 2$, so $\pi^{-1}(2) \in \{I_2, I_3\}$;
- $\pi^{-1}(3) \triangleleft 3$, so $\pi^{-1}(3) \in \{I_1, I_2\}$.

Three bijections satisfy these constraints:

$$\begin{aligned} \pi_1: I_1 \mapsto H, I_2 \mapsto 3, I_3 \mapsto 2, \\ \pi_2: I_1 \mapsto 3, I_2 \mapsto 2, I_3 \mapsto H, \\ \pi_3: I_1 \mapsto 3, I_2 \mapsto H, I_3 \mapsto 2. \end{aligned}$$

6.3.1. *Realizable and unrealizable bijections.* The bijections π_1 and π_2 are *realizable*: each admits successful castings arising from performances with different temporal orderings (Figures 12 and 13). Both have the same permutation sign, $\text{sgn}(\pi_1) = \text{sgn}(\pi_2) = \text{sgn} \mathcal{F} = -1$, as guaranteed by Proposition 6.1.

The bijection π_3 is *unrealizable*: it satisfies the candidacy constraint but admits no successful casting (Figure 15). The middle particle I_2 must coalesce with a neighbor before reaching the final time. If I_2 coalesces first with I_1 , then ghost 2 is created—but the ghost must be I_1 or I_2 , not I_3 as π_3 requires. If I_2 coalesces first with I_3 , then ghost 3 is created—but the ghost must be I_2 or I_3 , not I_1 as π_3 requires.

6.3.2. *The involution pairs failed castings.* Every π_3 -casting fails at its first crossing (either $P_1 \cap P_2$ or $P_2 \cap P_3$). The segment swap at this crossing converts π_3 -castings into π_1 - or π_2 -castings with wrong crossing order. Conversely, π_1 - and π_2 -castings with wrong crossing order swap back to π_3 -castings. The pairings preserve weight and flip sign, so all failed castings cancel. Only correctly-ordered π_1 - and π_2 -castings survive—exactly those arising from performances.

7. CONTINUOUS PROCESSES

The discrete framework extends to continuous time and space. This section proves that the ghost formulas hold for processes satisfying the Karlin–McGregor assumptions; concrete examples (Brownian motion, simple random walk) appear in Section 8.4.

7.1. Karlin–McGregor assumptions. Let Ω denote the probability space for n non-interacting particles with initial positions $x_1 \leq \dots \leq x_n$. Each $\omega \in \Omega$ specifies trajectories up to time T , with $\mathbf{X}_T(\omega) = (X_T^1(\omega), \dots, X_T^n(\omega))$ denoting the final positions.

The *Karlin–McGregor assumptions* [KM59] are:

- (1) *Identical dynamics*: all particles follow the same Markov process (differing only in initial position);
- (2) *Order preservation*: adjacent particles cannot change their relative order without first occupying the same state;
- (3) *Strong Markov property*: for any stopping time τ , the post- τ process is conditionally independent of the pre- τ process given the state at τ ;
- (4) *Meeting times are stopping times*: for particles $I < J$, the first meeting time $\tau_{I,J} = \inf\{t : X_t^I = X_t^J\}$ is a stopping time.

Assumption (1) ensures that when two particles meet, their post-meeting trajectories are exchangeable (conditioned on the meeting state): this is the measure-theoretic analog of weight-preserving segment swap. These hold for Brownian motion and other continuous-path diffusions, as well as for skip-free birth-death chains. For discrete-time ± 1 walks on \mathbb{Z} , order preservation requires that all particles share the same parity (as enforced by the checkerboard lattice of Example 2.3).

7.2. The continuous-time ghost formula.

Definition 7.1 (Admissible sets). An *admissible set* $A \subseteq \mathbb{R}^{\mathcal{R}}$ is a measurable set of final positions consistent with a fixed sign vector ε : for each ghost g , every $(y_f)_{f \in \mathcal{R}} \in A$ satisfies $y_g \preceq y_{\text{heir}(g)}$ if $\varepsilon(g) = +1$ and $y_{\text{heir}(g)} \prec y_g$ if $\varepsilon(g) = -1$. The sign $\text{sgn}(A) = \prod_g \varepsilon(g)$ and candidate set Π_A (Definition 4.3) depend only on the final entities and ε , not on specific positions in A .

Theorem 7.2 (Continuous-time ghost formula). *Let the underlying process satisfy the Karlin–McGregor assumptions. For any admissible set A :*

$$\mathbb{P}_{\text{int}}(\text{final positions} \in A) = \text{sgn}(A) \sum_{\pi \in \Pi_A} \text{sgn}(\pi) \mathbb{P}(\mathbf{X}_T \in A_\pi),$$

where \mathbb{P}_{int} denotes the probability for the coalescing system, \mathbb{P} denotes the probability for non-interacting particles, and

$$A_\pi = \{(y_{\pi(1)}, \dots, y_{\pi(n)}) : (y_1, \dots, y_n) \in A\}$$

is the set A with coordinates permuted by π .

The formula is an identity between measures; it makes no reference to densities or mass functions. The continuous-time formula is the basis for the Brownian motion specialization in [Sni26a], where scaling limits of the coalescence determinant yield explicit gap distributions for coalescing Brownian motions, and in [Sni26c], where it underlies the derivation of Pfaffian point process structure.

7.3. Proof.

Proof of Theorem 7.2. The proof parallels the discrete argument of Sections 4 to 6. Each step has a measure-theoretic analog: castings become elements of a product measure space, the sign-reversing involution becomes a measure-preserving map, and the cancellation of failed castings follows from exchangeability of post-meeting trajectories (via the strong Markov property).

Casting space.

Definition 7.3 (Casting space). For an admissible set A , the *casting space* is

$$\mathcal{C}_A = \{(\pi, \omega) \in \Pi_A \times \Omega : \mathbf{X}_T(\omega) \in A_\pi\}.$$

A *casting* $(\pi, \omega) \in \mathcal{C}_A$ pairs a candidate bijection with n trajectories $(P_I)_{I \in A}$; its *sign* is $\text{sgn}(\pi, \omega) = \text{sgn}(\pi)$. The casting space carries the product measure μ : counting measure on Π_A times the probability measure \mathbb{P} on Ω .

Partition of the casting space. The involution $\iota : \mathcal{C}_A \rightarrow \mathcal{C}_A$ acts by segment swap at the first spurious crossing. Partition the casting space according to which pair of actors has the first spurious crossing:

$$\mathcal{C}_A = \mathcal{C}_A^{\text{succ}} \sqcup \bigsqcup_{I < J} \mathcal{C}_A^{(I, J)},$$

where:

- $\mathcal{C}_A^{\text{succ}}$ consists of *successful* castings (no spurious crossing); the involution acts as the identity;
- $\mathcal{C}_A^{(I, J)}$ consists of castings whose first spurious crossing involves actors I and J ; the involution equals the local involution $\iota_{I, J}$ (segment swap for paths P_I, P_J).

Measure-preservation of local involutions. Each local involution $\iota_{I, J}$ is measure-preserving. At the meeting time $\tau_{I, J}$, particles I and J occupy the same position. By the strong Markov property, conditioned on the common state at $\tau_{I, J}$ and on the pre- $\tau_{I, J}$ trajectories, the post- $\tau_{I, J}$ trajectories of I and J are conditionally independent and identically distributed. In particular, the joint law of the pair $(P_I^{\text{post}}, P_J^{\text{post}})$ is exchangeable: swapping the two post-meeting segments produces a pair with the same distribution. Since $\iota_{I, J}$ acts by exchanging exactly these segments while leaving all other paths unchanged, it preserves the product measure μ .

Cancellation of failed castings. On the failed part $\mathcal{C}_A \setminus \mathcal{C}_A^{\text{succ}}$, the involution ι is sign-reversing: $\text{sgn}(\iota(\pi, \omega)) = -\text{sgn}(\pi)$. This holds because the segment swap exchanges the roles of actors I and J , so the new bijection $\pi' = (I \ J) \circ \pi$ differs from π by a transposition.

Since ι is also measure-preserving:

$$\int_{\mathcal{C}_A \setminus \mathcal{C}_A^{\text{succ}}} \text{sgn}(\pi) d\mu = \int_{\mathcal{C}_A \setminus \mathcal{C}_A^{\text{succ}}} \text{sgn}(\iota(\pi, \omega)) d\mu = - \int_{\mathcal{C}_A \setminus \mathcal{C}_A^{\text{succ}}} \text{sgn}(\pi) d\mu.$$

Therefore $\int_{\text{failed}} \text{sgn}(\pi) d\mu = 0$, and

$$\int_{\mathcal{C}_A} \text{sgn}(\pi) d\mu = \int_{\mathcal{C}_A^{\text{succ}}} \text{sgn}(\pi) d\mu.$$

Contribution from successful castings. For successful castings, the sign identity (Proposition 6.1) gives $\text{sgn}(\pi) = \text{sgn}(A)$. Therefore

$$\int_{\mathcal{C}_A^{\text{succ}}} \text{sgn}(\pi) d\mu = \text{sgn}(A) \cdot \mu(\mathcal{C}_A^{\text{succ}}) = \text{sgn}(A) \cdot \mathbb{P}_{\text{int}}(\text{final positions} \in A),$$

where the last equality uses the bijection between successful castings and performances (Proposition 5.15). The bijection is purely combinatorial—it depends only on the paths, not on the probability measure—so the discrete proof applies verbatim to each outcome ω : for each $\omega \in \Omega$ with final positions in A , there is exactly one $\pi \in \Pi_A$ such that (π, ω) is a successful casting.

Conclusion. The integral over the casting space equals

$$\int_{\mathcal{C}_A} \text{sgn}(\pi) d\mu = \sum_{\pi \in \Pi_A} \text{sgn}(\pi) \mathbb{P}(\mathbf{X}_T \in A_\pi).$$

Combining with the cancellation and sign identity results:

$$\text{sgn}(A) \cdot \mathbb{P}_{\text{int}}(\text{final positions} \in A) = \sum_{\pi \in \Pi_A} \text{sgn}(\pi) \mathbb{P}(\mathbf{X}_T \in A_\pi).$$

Multiplying both sides by $\text{sgn}(A)$ yields Theorem 7.2. \square

8. INTEGRATING OUT THE GHOSTS

The coalescence formula (Theorem 3.2) tracks ghost positions via formal variables. For applications, one often wants the probability of survivor positions alone. This section integrates out the ghost variables to obtain a closed-form determinantal formula—the *coalescence determinant*—that, like Theorem 7.2, is an identity between measures, valid for both discrete and continuous state spaces. The matrix definition and theorem statement use only the coalescence pattern and transition kernels; the ghost machinery enters only in the derivation.

8.1. Transition kernels. Let S denote the state space (\mathbb{R} or \mathbb{Z}), and fix a time horizon $T > 0$. Write X_T for the position of a single particle at time T , and let P_x denote the transition kernel: the distribution of X_T when the particle starts at x . For a measurable set B , write $P_x(B) = \mathbb{P}_x(X_T \in B)$. Let ν denote the reference measure (Lebesgue measure on \mathbb{R} , counting measure on \mathbb{Z}), and write $p_x(y)$ for the density of P_x with respect to ν : the transition density (continuous case) or transition probability (discrete case). Define the cumulative distribution

$$F_x(y) = P_x((-\infty, y]) = \mathbb{P}_x(X_T \leq y).$$

8.2. The coalescence matrix. Fix a coalescence pattern: a composition $c_1 + \cdots + c_k = n$, where the first c_1 particles merge into survivor 1 at position y_1 , the next c_2 into survivor 2 at y_2 , and so on. The l th block of the composition—the initial particles merging into survivor l —has indices $c_1 + \cdots + c_{l-1} + 1$ through $c_1 + \cdots + c_l$.

Definition 8.1 (Coalescence matrix). Both rows and columns of the $n \times n$ *coalescence matrix* \tilde{M} are indexed by $\{1, \dots, n\}$. The entry in row i , column j (where j lies in the l th block, with survivor position y_l) is

$$\tilde{M}_{ij} = \begin{cases} p_{x_i}(y_l) & \text{if } j \text{ is the first index in its block,} \\ F_{x_i}(y_l) - [i < j] & \text{otherwise,} \end{cases}$$

where $[i < j]$ denotes the Iverson bracket. The first column of each block contains transition densities (or probabilities); the remaining $c_l - 1$ columns contain cumulative distributions with a “staircase” shift $-[i < j]$: entries with $i < j$ are shifted by -1 (see Figure 2).

8.3. The coalescence determinant.

Theorem 8.2 (Coalescence determinant). *Let $A \subseteq S^k$ be a measurable set of survivor positions (y_1, \dots, y_k) . Then:*

$$\mathbb{P}_{\text{int}}(\text{survivor positions} \in A) = \int_A \det(\tilde{M}(y_1, \dots, y_k)) d\nu,$$

where \mathbb{P}_{int} denotes the probability for the coalescing system (Section 7).

The formula is an identity between measures: $\det(\tilde{M})$ is the Radon–Nikodym derivative of the survivor-position distribution with respect to ν . For continuous state spaces, $\det(\tilde{M})$ is a probability density; for discrete state spaces, it is a probability mass function. The companion papers [Śni26a] and [Śni26c] develop applications of the coalescence determinant.

Proof. Start from Theorem 3.2: the generating function for a fixed sign vector ε is $Z_\varepsilon = [\prod_g t_g^{\varepsilon(g)}] \det(M)$. To marginalize over ghost positions, integrate each ghost position y_g over the state space S and sum over all sign vectors $\varepsilon \in \{+1, -1\}^{\mathcal{G}}$.

Using multilinearity of the determinant in columns, the summation and integration act column by column. Heir columns contain $W(x_i \rightarrow y_H)$ with no formal variables, so they pass through unchanged; under continuous state spaces, $W(x_i \rightarrow y_H)$ becomes the transition density $p_{x_i}(y_H)$.

For a ghost column g with heir $H = \text{heir}(g)$ and rank $r = \text{rank}(g)$, the matrix entry is $t_g^+ \cdot W(x_i \rightarrow y_g)$ when $i \geq r$ and $-t_g^- \cdot W(x_i \rightarrow y_g)$ when $i < r$. Coefficient extraction $[t_g^{\varepsilon(g)}]$ selects one term per row: the coefficient of the matching formal variable (and 0 when the variable is absent). Summing over both signs $\varepsilon(g) \in \{+1, -1\}$ and integrating over y_g , the ghost column entry in row i becomes:

- If $i \geq r$: the entry $t_g^+ \cdot W(x_i \rightarrow y_g)$ contributes when $\varepsilon(g) = +1$ (ghost left of heir, $y_g \leq y_H$). Integrating:

$$\int_{y_g \leq y_H} p_{x_i}(y_g) d\nu(y_g) = F_{x_i}(y_H).$$

- If $i < r$: the entry $-t_g^- \cdot W(x_i \rightarrow y_g)$ contributes when $\varepsilon(g) = -1$ (ghost right of heir, $y_g > y_H$). The coefficient of t_g^- is $-W(x_i \rightarrow y_g)$. Integrating:

$$-\int_{y_g > y_H} p_{x_i}(y_g) d\nu(y_g) = -(1 - F_{x_i}(y_H)) = F_{x_i}(y_H) - 1.$$

This is precisely the staircase pattern in Definition 8.1. □

8.4. Example.

Example 8.3 (Pattern 2+1: three particles, first two coalesce). Continuing Section 1.3.2: particles 1 and 2 coalesce (survivor at y_1) while particle 3 survives alone (y_2). The composition is 2+1: the first block has columns 1 and 2, the second block

has column 3. The coalescence matrix is:

$$\tilde{M} = \begin{pmatrix} p_{x_1}(y_1) & F_{x_1}(y_1) - 1 & p_{x_1}(y_2) \\ p_{x_2}(y_1) & F_{x_2}(y_1) & p_{x_2}(y_2) \\ p_{x_3}(y_1) & F_{x_3}(y_1) & p_{x_3}(y_2) \end{pmatrix}.$$

Column 1 (first in its block) contains transition densities. Column 2 (cumulative) has the staircase shift: row 1 has $i < j$, giving $F - 1$; rows 2 and 3 have $i \geq j$, giving F . Column 3 (first in the second block) again contains densities.

For any measurable set $A \subseteq S^2$ of survivor positions:

$$\mathbb{P}_{\text{int}}((y_1, y_2) \in A) = \int_A \det(\tilde{M}) d\nu.$$

For Brownian motion ($S = \mathbb{R}$), $d\nu = dy_1 dy_2$ and $\det(\tilde{M})$ is a joint density. For simple random walk ($S = \mathbb{Z}$), ν is counting measure and $\det(\tilde{M})$ is the probability mass at (y_1, y_2) .

ACKNOWLEDGMENTS

We thank Theodoros Assiotis, Balázs Bárány, Maciej Dołęga, Sho Matsumoto, Bálint Tóth, Oleg Zaboronski, and Karol Życzkowski for stimulating discussions and helpful literature suggestions.

Claude Code (Anthropic) was used by the first-named author as an assistant during formula discovery and manuscript preparation.

REFERENCES

- [AB05] Daniel ben Avraham and Éric Brunet. “On the relation between one-species diffusion-limited coalescence and annihilation in one dimension”. In: *J. Phys. A: Math. Gen.* 38 (2005), pp. 3247–3252. DOI: [10.1088/0305-4470/38/15/001](https://doi.org/10.1088/0305-4470/38/15/001). arXiv: [cond-mat/0412745](https://arxiv.org/abs/cond-mat/0412745).
- [AOW19] Theodoros Assiotis, Neil O’Connell, and Jon Warren. “Interlacing Diffusions”. In: *Séminaire de Probabilités L*. Vol. 2252. Lecture Notes in Mathematics. Cham: Springer, 2019, pp. 301–380. DOI: [10.1007/978-3-030-28535-7_13](https://doi.org/10.1007/978-3-030-28535-7_13). arXiv: [1607.07182](https://arxiv.org/abs/1607.07182).
- [Arr79] Richard Arratia. “Coalescing Brownian motions on the line”. PhD thesis. University of Wisconsin–Madison, 1979.
- [Ass18] Theodoros Assiotis. “Random surface growth and Karlin–McGregor polynomials”. In: *Electron. J. Probab.* 23 (2018), Paper no. 106, 81 pp. DOI: [10.1214/18-EJP236](https://doi.org/10.1214/18-EJP236). arXiv: [1709.10444](https://arxiv.org/abs/1709.10444).
- [Ass23] Theodoros Assiotis. *On some integrable models in inhomogeneous space*. 2023. arXiv: [2310.18055](https://arxiv.org/abs/2310.18055) [[math.PR](https://arxiv.org/abs/math.PR)].
- [Avr98] Daniel ben Avraham. “Complete exact solution of diffusion-limited coalescence, $A + A \rightarrow A$ ”. In: *Phys. Rev. Lett.* 81 (1998), pp. 4756–4759. DOI: [10.1103/PhysRevLett.81.4756](https://doi.org/10.1103/PhysRevLett.81.4756). arXiv: [cond-mat/9803281](https://arxiv.org/abs/cond-mat/9803281).
- [BŠTU26] Balázs Bárány, Piotr Śniady, Bálint Tóth, and Ákos Urbán. “The Pólya Web”. In preparation. 2026.
- [DA88] Charles R. Doering and Daniel ben Avraham. “Interparticle distribution functions and rate equations for diffusion-limited reactions”. In: *Phys. Rev. A* 38 (1988), pp. 3035–3042. DOI: [10.1103/PhysRevA.38.3035](https://doi.org/10.1103/PhysRevA.38.3035).

- [Fom16] Vladimir Fomichov. “The distribution of the number of clusters in the Arratia flow”. In: *Communications on Stochastic Analysis* 10.3 (2016), pp. 257–270. DOI: [10.31390/cosa.10.3.01](https://doi.org/10.31390/cosa.10.3.01).
- [GF17] E. V. Glinyanaya and V. V. Fomichov. “The central limit theorem for the number of clusters of the Arratia flow”. In: *Theory of Stochastic Processes* 22(38).2 (2017), pp. 1–7. arXiv: [1712.05098](https://arxiv.org/abs/1712.05098).
- [GM81] Adriano M. Garsia and Stephen C. Milne. “A Rogers–Ramanujan bijection”. In: *J. Combin. Theory Ser. A* 31.3 (1981), pp. 289–339. DOI: [10.1016/0097-3165\(81\)90062-5](https://doi.org/10.1016/0097-3165(81)90062-5).
- [GPTZ18] Barnaby Garrod, Mihail Poplavskiy, Roger Tribe, and Oleg Zaboronski. “Examples of interacting particle systems on \mathbb{Z} as Pfaffian point processes: annihilating and coalescing random walks”. In: *Ann. Henri Poincaré* 19 (2018), pp. 3635–3662. DOI: [10.1007/s00023-018-0719-x](https://doi.org/10.1007/s00023-018-0719-x).
- [GV85] Ira Gessel and Gérard Viennot. “Binomial determinants, paths, and hook length formulae”. In: *Adv. Math.* 58 (1985), pp. 300–321. DOI: [10.1016/0001-8708\(85\)90121-5](https://doi.org/10.1016/0001-8708(85)90121-5).
- [HL75] Richard A. Holley and Thomas M. Liggett. “Ergodic theorems for weakly interacting infinite systems and the voter model”. In: *Ann. Probab.* 3.4 (1975), pp. 643–663. DOI: [10.1214/aop/1176996306](https://doi.org/10.1214/aop/1176996306).
- [KM59] Samuel Karlin and James McGregor. “Coincidence probabilities”. In: *Pacific J. Math.* 9.4 (1959), pp. 1141–1164. DOI: [10.2140/pjm.1959.9.1141](https://doi.org/10.2140/pjm.1959.9.1141).
- [Kra15] Christian Krattenthaler. “Lattice path enumeration”. In: *Handbook of Enumerative Combinatorics* (2015), pp. 589–678. DOI: [10.1201/b18255-13](https://doi.org/10.1201/b18255-13). arXiv: [1503.05930](https://arxiv.org/abs/1503.05930).
- [Lin73] Bernt Lindström. “On the vector representations of induced matroids”. In: *Bull. London Math. Soc.* 5 (1973), pp. 85–90. DOI: [10.1112/blms/5.1.85](https://doi.org/10.1112/blms/5.1.85).
- [Śni26a] Piotr Śniady. “Coalescing random walks via the coalescence determinant”. Preprint. 2026. arXiv: [2602.20043](https://arxiv.org/abs/2602.20043) [[math.PR](https://arxiv.org/abs/2602.20043)].
- [Śni26b] Piotr Śniady. “Determinant and Pfaffian formulas for particle annihilation”. Preprint. 2026. arXiv: [2602.13183](https://arxiv.org/abs/2602.13183) [[math.PR](https://arxiv.org/abs/2602.13183)].
- [Śni26c] Piotr Śniady. “Pfaffian structure of basin walls for coalescing particles”. Preprint. 2026. arXiv: [2602.22885](https://arxiv.org/abs/2602.22885) [[math.PR](https://arxiv.org/abs/2602.22885)].
- [Ste90] John R. Stembridge. “Nonintersecting paths, Pfaffians, and plane partitions”. In: *Adv. Math.* 83 (1990), pp. 96–131. DOI: [10.1016/0001-8708\(90\)90070-4](https://doi.org/10.1016/0001-8708(90)90070-4).
- [ŚU26] Piotr Śniady and Ákos Urbán. *Companion code for “Exact determinant formulas for coalescing particle systems”*. Python package, <https://psniady.impan.pl/kmg>. 2026.
- [TZ11] Roger Tribe and Oleg Zaboronski. “Pfaffian formulae for one-dimensional coalescing and annihilating systems”. In: *Electron. J. Probab.* 16 (2011), pp. 2080–2103. DOI: [10.1214/EJP.v16-942](https://doi.org/10.1214/EJP.v16-942).
- [TZ26] Roger Tribe and Oleg Zaboronski. “Entrance laws for coalescing and annihilating Brownian motions”. Preprint. 2026. arXiv: [2602.16509](https://arxiv.org/abs/2602.16509) [[math.PR](https://arxiv.org/abs/2602.16509)].
- [Urb25] Ákos Urbán. “The Pólya Web”. MA thesis. Budapest University of Technology and Economics, 2025. arXiv: [2601.12172](https://arxiv.org/abs/2601.12172) [[math.PR](https://arxiv.org/abs/2601.12172)].

- [War07] Jon Warren. “Dyson’s Brownian motions, intertwining and interlacing”. In: *Electron. J. Probab.* 12 (2007), pp. 573–590. DOI: [10.1214/EJP.v12-406](https://doi.org/10.1214/EJP.v12-406).

(P. Śniady) INSTITUTE OF MATHEMATICS, POLISH ACADEMY OF SCIENCES, UL. ŚNIADECKICH 8, 00-656 WARSZAWA, POLAND
Email address, P. Śniady: psniady@impan.pl

(Á. Urbán) DEPARTMENT OF STOCHASTICS, BUDAPEST UNIVERSITY OF TECHNOLOGY AND ECONOMICS, BUDAPEST, HUNGARY
Email address, Á. Urbán: urbana@math.bme.hu