Forgetting Event Order in Higher-Dimensional Automata

Safa Zouari
Norwegian University of Science and Technology, Gjøvik, Norway

Abstract

Higher-dimensional automata (HDAs) provide a geometric model of true concurrency, yet their standard formulation encodes an artificial total order on events. This representational artifact causes a fundamental mismatch between the combinatorial structure of HDAs and their observable behaviour, leading to logical asymmetries and complicating the application of categorical tools. In this paper, we resolve this tension by developing a semantics for HDAs that is independent of event order, based on interval ipomsets (partially ordered multisets with interfaces) that preserve only precedence and concurrency. We prove that for any HDA, the traditional ST–trace of an execution path corresponds precisely to its associated interval ipomset. On the structural side, we show that the presheaf-theoretic presentation with an unordered base and the combinatorial presentation of symmetric HDAs are categorically isomorphic. Finally, by characterizing ST- and hereditary history-preserving (hhp) bisimulation via ipomset isomorphism, we provide a unified, order-free foundation for HDA semantics. Our results resolve several ambiguities in the literature: they provide the necessary path-category structure to canonically apply the Open Maps framework, eliminate representational artifacts in temporal and modal logics, and bridge systematic mismatches between HDAs and other models of concurrency such as Petri nets.

2012 ACM Subject Classification Theory of computation → Concurrency; Theory of computation → Automata extensions

Keywords and phrases higher-dimensional automata, true concurrency, pomsets with interfaces, interval pomsets, pomsets, presheaf semantics, bisimulation, event order, ST–trace

Digital Object Identifier 10.4230/LIPIcs.LICS.2026.84

Funding *Safa Zouari*: Supported by a PhD fellowship at the Department of Information Security and Communication Technology (IIK), Norwegian University of Science and Technology (NTNU).

Acknowledgements I thank Krzysztof Ziemiański and Rob van Glabbeek for helpful comments and insightful discussions. Any remaining errors are mine alone.

1 Introduction

Higher-dimensional automata (HDAs) offer a geometric view of true concurrency: higher-dimensional cells represent independent events occurring simultaneously rather than through artificial interleaving. This structure makes HDAs a unifying semantic framework for concurrency, as Petri nets, event structures, and asynchronous transition systems all embed into HDAs up to hereditary history-preserving bisimulation [19, 23, 18, 24, 7, 21].

In existing work, pomset-based semantics impose an extraneous event order inherited from the HDA's combinatorial structure. This order on conclists is a positional index: it ranks concurrent events by their place in a list, but the ranking has no counterpart in any observation, including ST-traces and Petri-net behaviours. Consider two processes executing actions a and b concurrently. An observer records that both started and both finished, with no ranking between them. ST-traces capture exactly this content: they record causal pairings between starts and terminations but impose no order among simultaneous events. Consequently, the event order causes HDAs to distinguish $a \parallel b$ from $b \parallel a$ even though no observation can tell them apart, so pomset labels frequently treat as distinct executions that are observationally equivalent. This mismatch has tangible consequences for logic. In temporal and modal logics for HDAs [2, 8, 3, 25], the imposed order breaks symmetry: a formula may accept $a \parallel b$ but reject $b \parallel a$, even though these executions are observationally indistinguishable.

Such asymmetries are artifacts of the representation rather than of the underlying logic. Moreover, injecting Petri-net behaviours into ordered HDA frameworks leads to systematic mismatches [6, 4], pointing to a deeper structural problem: the standard event ordered representation is fundamentally incompatible with the categorical tools used to reason about concurrency. Specifically, while the Open Maps framework [16] provides a standard recipe for deriving modal logics that characterize *hereditary history preserving bisimulation* (hhp-bisimulation), its application to HDAs has remained ambiguous. The reliance on ST-traces lacks the necessary path-category structure, leaving the relationship between observational traces and categorical logic poorly defined.

Two distinct notions of event order appear in the HDA literature, and they have different histories. The first is the order built into the *combinatorial definition* of HDAs, which assigns distinct combinatorial cells to $(a \parallel b)$ and $(b \parallel a)$ even though no observation can tell them apart. The second is the order carried by *ipomset labels* assigned to paths in the pomset semantics of HDAs.

For the first, two responses exist in the literature. Kahl extended the combinatorial structure with permutation morphisms to obtain symmetric HDAs [17]. Struth and Ziemiański took the dual route of dropping the order from the base category, working over an unordered presheaf site [22], and conjectured that the two presentations describe the same class of HDAs. The conjecture was not proven. Section 2 and Section 3 settle it: we establish a categorical isomorphism between the unordered presheaf base and Kahl's symmetric base, so the two approaches coincide. This is what allows us to import Kahl's Symmetrization Theorem into the presheaf framework and obtain order-free semantics for *all* HDAs, not only the symmetric ones.

The second has a more layered history. Fahrenberg et al. [9] introduced gluing composition for ipomsets as an algebraic operation, independently of HDAs: their construction places no order on the pomset carrier and uses a canonicality condition on the *interfaces* to make gluing well-defined. When the path-labelling theory of HDAs was developed in [10], an event order on the carrier was added to control gluing in full generality; within [10] this choice is internally consistent and serves a real purpose. The same choice, however, becomes a source of friction downstream: work building on [10], including the temporal and modal logics of [2, 8, 3, 25] and the language-theoretic developments of [5, 11], inherits the carrier-level order and with it a residual asymmetry: a formula can accept $a \parallel b$ but reject $b \parallel a$, and language-theoretic statements about HDA paths have to carry the order through. These executions are observationally indistinguishable, so the asymmetry is an artifact of the labelling rather than of the underlying behaviour.

Our contribution on this side starts from the gluing of [9] but goes further. We work in the order-free setting it makes available and define isomorphism of ipomsets directly there (Definition 23), without the event-order constraint that ordered approaches require their isomorphisms to preserve. We then take ipomset *isomorphism classes* as the labels of HDA paths (Section 6) and prove that this choice is canonical in a precise sense: each isomorphism class corresponds exactly to an ST-trace of the path (Section 7), the established observational semantics of HDAs introduced by van Glabbeek [23]. This is the argument that the order-free presentation is not merely available but correct: ipomset-isomorphism classes are the natural observable content of HDA paths, because they recover the trace semantics that the field has long accepted as canonical. The downstream effect is that the residual asymmetries in the logics and languages built on [10] dissolve: $a \parallel b$ and $b \parallel a$ become genuinely isomorphic ipomsets, and the modal and temporal logics for HDAs can be revisited on a foundation that no longer carries the artifact.

We summarize our contributions below.

- **Categorical isomorphism (Sections 2–3):** HDAs over the unordered presheaf base of [22] are categorically isomorphic to the symmetric HDAs of [17]. This imports Kahl's Symmetrization Theorem into the presheaf framework, so order-free semantics applies to all HDAs.
- **Order-free path labels (Section 6):** A canonical interval ipomset is assigned to each HDA path via a labelling functor, intrinsic to the execution and free of representational artifacts.
- **Trace correspondence (Section 7):** Ipomset isomorphism classes correspond exactly to ST-traces, identifying ipomsets as the faithful semantic counterpart to the established observational semantics.
- **Bisimulation and Open Maps (Section 8):** ST- and hhp-bisimulation are characterized via ipomset isomorphism. The order-free path-category structure provides the input the Open Maps framework requires, resolving the previous ambiguity in deriving modal logics that characterize hhp-bisimulation.

2 Base categories

This section introduces the base categories underlying our variants of higher-dimensional automata. They differ in how they represent *event ordering* and *symmetry*. The category $\square$ captures ordered concurrency through labeled event lists, while Ξ omits the event order, retaining only labels. The generator based version Ξ_g extends $\square$ by freely adding permutation morphisms. We prove that $\Xi_g \cong \Xi$, showing that adding event symmetries is equivalent to removing the event order. After establishing this equivalence, we identify Ξ_g with Ξ.

2.1 The labeled precube category $\square$

Let Σ be a given set of *actions*.

▶ **Definition 1** (Concurrency list, [22]). *A* concurrency list *(or* conclist*) is a tuple $(U, \dashrightarrow, \lambda)$ where U is a finite set equipped with a strict (i.e., irreflexive) total order $\dashrightarrow$, called the event order. The function $\lambda : U \to \Sigma$ is a labelling map.*

The set U models the concurrent local events active in a cell of an HDA, whereas the event order $\dashrightarrow$ can be seen as their index order.

▶ **Definition 2** ([1]). *A* conclist map *from a conclist U to V is a pair (f, ε) such that:*

- *$f : U \to V$ is a label and order-preserving function;*
- *$\varepsilon : V \to \{0, \lrcorner, 1\}$ is a function such that $\varepsilon^{-1}(\lrcorner) = f(U)$.*

The composition of morphisms $(f, \varepsilon) : S \to T$ and $(g, \zeta) : T \to U$ is $(g, \zeta) \circ (f, \varepsilon) = (g \circ f, \eta)$, where $\eta(u) = \begin{cases} \varepsilon(t) & \text{if } u = g(t) \text{ for some } t \in T, \\ \zeta(u) & \text{if } u \notin g(T). \end{cases}$

The full labeled precube category *$\square_{full}$ has conclists as objects and conclist maps as morphisms.*

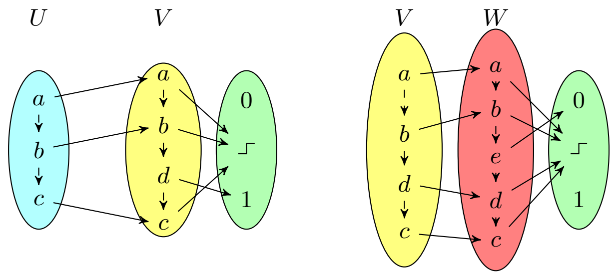

Figure 1 Example of conclist maps $(f, \varepsilon) : U \to V$ (on the left) and $(g, \zeta) : V \to W$ (on the right).

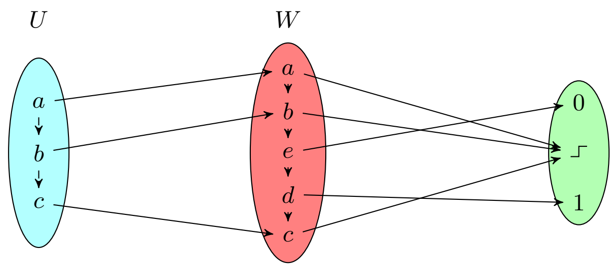

Figure 2 Illustration of $(g, \zeta) \circ (f, \varepsilon) : U \to W$, the composition of the conclist maps of Fig 1.

The *restricted* category $\square$ is the full subcategory on canonical conclists (n, λ) with carrier $\{1 \dashrightarrow \cdots \dashrightarrow n\}$ (and $(0) = \emptyset$). The inclusion $\square \hookrightarrow \square_{full}$ is an equivalence; hence their presheaf categories are naturally equivalent [10].

Intuition. Fix a conclist map $(f, \varepsilon) : U \to V$ with $|U| = m$, $|V| = n$ $(m \leq n)$. Viewing V as the list of n potential concurrent events, ε classifies each $v \in V$ as *not yet started* (0), *executing* ($\lrcorner\ulcorner$), or *terminated* (1). The map f specifies how the m active events of U appear (by name and order) inside V; thus f is determined by the inactive set $V \setminus \varepsilon^{-1}(\lrcorner\ulcorner)$. See Fig 1 and 2 for examples of conclist maps and their composition.

Isomorphisms. A conclist map $(f, \varepsilon) : U \to V$ is an isomorphism iff f is bijective; then necessarily $\varepsilon \equiv \lrcorner\ulcorner$. We write $U \cong V$ for isomorphic conclists. If two conclists are isomorphic, then the isomorphism between them is unique [11].

Insertion maps ι_i. For canonical conclists $U = [n-1, \lambda]$ and $V = [n, \lambda']$, let $\iota_i : U \hookrightarrow V$ $(1 \leq i \leq n)$ be the unique order- and label-preserving injection whose image omits the i-th element of V. It has the following explicit expression

$$\iota_i(j) = \begin{cases} j & j < i, \\ j + 1 & j \geq i. \end{cases}$$

Canonical coface maps. For canonical conclists $U = (n–1, \lambda)$ and $V = (n, \lambda')$ and $k \in \{0, 1\}$, the coface map $d^k_{i,n} : U \to V$ $(1 \leq i \leq n)$ is the unique conclist map (ι_i, ε) such that ι_i is the insertion map, $\varepsilon(i) = k$, and $\varepsilon(j) = \lrcorner\ulcorner$ $(j \neq i)$. When n is clear we write d^k_i.

2.2 The symmetric labeled precube category Ξ

▶ **Definition 3** (Concurrency set). *A* concurrency set *(or* concset*) is a pair* (U, λ) *where* U *is a finite set and* $\lambda : U \to \Sigma$ *is a labelling map. That is, concsets are just conclists without an event order.*

A *concset map* from U to V is a pair (f, ε) such that

- $f : U \to V$ is an injective, label-preserving function;
- $\varepsilon : V \to \{0, \lrcorner, 1\}$ satisfies $\varepsilon^{-1}(\lrcorner) = f(U)$.

Composition of concset maps is defined as in Definition 2.

Write Ξ_{full} for the full symmetric labeled precube category on all concsets and concset maps. The *restricted* subcategory Ξ has as objects the canonical concsets (n, λ) with carrier $\{1, \dots, n\}$ (and $(0) = \emptyset$). As previously, the inclusion $\Xi \hookrightarrow \Xi_{full}$ is an equivalence.

Interpretation. A similar interpretation applies here; given a concset map $(f, \varepsilon) : U \to V$ with $|U| = m$ and $|V| = n$ $(m \leq n)$, the function $\varepsilon : V \to \{0, \lrcorner, 1\}$ classifies each event of V as *not yet started* (0), *executing* ($\lrcorner$), or *terminated* (1), and $f : U \hookrightarrow V$ embeds the m active events by ensuring $\varepsilon^{-1}(\lrcorner) = f(U)$. However, unlike the ordered case $\square$, the absence of an event order in Ξ means that ε does not need to determine f.

Example. From $(2, aa)$ to $(3, aab)$, take $\varepsilon = (\lrcorner, \lrcorner, 0)$ on the target. There are two distinct label-preserving injections f_1, f_2 that map the two a's of the source to the two a's of the target (identity vs. swap). Thus the same ε admits multiple f's. Consequently, concset isomorphisms are not unique; in fact, any permutation of equally labeled events yields another isomorphism.

Forgetting order. There is an obvious forgetful functor $s : \square \to \Xi$ that erases the event order on objects and sends a conclist map (f, ε) to the same pair viewed as a concset map. Thus $\square$ is a wide subcategory of Ξ. The inclusion is not full (e.g. nontrivial permutations exist in Ξ but not in $\square$). We henceforth identify $\square$ with its image under s and write $\square \subseteq \Xi$.

Canonical coface maps. For canonical concsets $U = (n-1, \gamma)$ and $V = (n, \gamma')$ we define analogously $d^k_{i,n} : U \to V$ with the same ε and insertion map ι_i (now without the event order).

2.3 The generator based categories Ξ_g and $\square_g$

This section introduces the base categories for our HDA variants, which differ in their treatment of *event ordering* and *symmetry*. While $\square$ represents ordered concurrency, Ξ omits event order to provide a coordinate-free foundation. We introduce Ξ_g by freely adding permutation morphisms to $\square$ and prove $\Xi_g \cong \Xi$. This isomorphism demonstrates that **adding symmetries is equivalent to removing event order**, allowing us to identify Ξ_g with Ξ henceforth.

Symmetric group and induced permutations on faces. For $n \geq 0$ let $\mathfrak{S}_n$ be the symmetric group on $\{1, \dots, n\}$; write $\mathfrak{S} = \bigsqcup_{n \geq 0} \mathfrak{S}_n$. For $n \geq 1$, $\theta \in \mathfrak{S}_n$, and $i \in \{1, \dots, n\}$, define the *induced permutation on faces* $d_i\theta \in \mathfrak{S}_{n-1}$ as the unique permutation making the square commute:

$$\begin{tikzcd} {[n-1]} \arrow[r, "d_i\theta"] \arrow[d, hook, "\iota_{\theta^{-1}(i)}"'] & {[n-1]} \arrow[d, hook, "\iota_i"] \\ {[n]} \arrow[r, "\theta"'] & {[n]} \end{tikzcd} \iff \iota_i \circ d_i\theta = \theta \circ \iota_{\theta^{-1}(i)}. \tag{1}$$

Intuitively, $d_i\theta$ is obtained from θ by deleting position $\theta^{-1}(i)$ in the domain and i in the codomain and renumbering. It can be expressed componentwise as

$$d_i\theta(j) = \begin{cases} \theta(j), & j < \theta^{-1}(i),\ \theta(j) < i, \\ \theta(j) - 1, & j < \theta^{-1}(i),\ \theta(j) > i, \\ \theta(j+1), & j \geq \theta^{-1}(i),\ \theta(j+1) < i, \\ \theta(j+1) - 1, & j \geq \theta^{-1}(i),\ \theta(j+1) > i. \end{cases}$$

▶ **Definition 4.** *The category Ξ_g is presented as follows.*

1. ***Objects.*** *Canonical conclists (n, λ) with $n \geq 0$ and a labelling map $\lambda : (n) \to \Sigma$.*
2. ***Generating morphisms.***
   a. Coface maps. *For each $n \geq 1$, $1 \leq i \leq n$, $k \in \{0, 1\}$ and label $\lambda : (n) \to \Sigma$, let $d_i^k : (n-1,\, \lambda \circ \iota_i) \to (n,\, \lambda)$,*
   b. Symmetry maps. *For each $n \geq 1$ and $\tau \in \mathfrak{S}_n$, let $\tau : (n,\, \lambda) \to (n,\, \lambda \circ \tau^{-1})$.*
3. ***Relations.***
   a. Cubical (face) identities. *For $i < j$, $d_j^\ell\, d_i^k = d_i^k\, d_{j-1}^\ell : (n-2,\, \lambda \circ \iota_i \circ \iota_{j-1}) \to (n,\, \lambda)$.*
   b. Group identities.[1] $\mathrm{id}_{(n,\lambda)}$ *is the identity and $\tau \circ \sigma = (\tau\sigma)$.*
   c. Permutation–face interchange. *For each $\tau \in \mathfrak{S}_n$ and $1 \leq i \leq n$, $d_i^k \circ d_i\theta = \theta \circ d_{\theta^{-1}(i)}^k$*

The category Ξ_g is obtained by freely adding identities and composites to these generators and quotienting by the stated relations.

We define the category $\square_g$ as the subcategory of Ξ_g obtained by forgetting about permutations, that is, by removing the symmetry generators and the corresponding relations. Concretely, $\square_g$ is defined by omitting Items 2(b), 3(b), and 3(c) in Definition 4.

2.4 Relation between the base categories

The goal of this paragraph is to establish relations between precubical categories we introduced so far and that we use in later sections. The goal is to show that these categories fit into the diagram

$$\begin{tikzcd} \square_{full} \arrow[d] & \square \arrow[l, "\supseteq"'] \arrow[r, "\cong"] \arrow[d, "\subseteq"] & \square_g \arrow[d] \\ \Xi_{full} & \Xi \arrow[l, "\supseteq"'] \arrow[r, "\cong"] & \Xi_g \end{tikzcd}$$

We want to prove that the left horizontal functors are natural equivalences (this part is fine) and that the right horizontal functors are isomorphisms. The latter allows us to identify Ξ and Ξ_g and use generators-and-relations and concset maps interchangeably.

[1] For brevity we often write simply id instead of $\mathrm{id}_{(n,\lambda)}$; the intended domain and codomain are always clear from the context.

Below is the construction of the functor $F : \Xi_g \to \Xi$ and the proof that it is an isomorphism.

▶ **Lemma 5.** *There exists a unique functor $F : \Xi_g \to \Xi$ such that*
- $F((n,\lambda)) = (n,\lambda)$ *(the identity on objects).*
- $F(d_i^k : (n,\lambda) \to (n-1, \lambda \circ \iota_i)) = (\iota_i, \varepsilon_i^k)$*, where* $\varepsilon_i^k(j) = k$ *if* $j = i$ *and* $\lrcorner$ *otherwise.*
- $F(\tau) = (\tau, const_{\lrcorner})$.

Proof. Since Ξ_g is presented by generators (face maps and permutations) and relations, it suffices to check that the assignment F preserves the defining relations.

Cubical relations. For $i < j$, the relation $d_j^\ell d_i^k = d_i^k d_{j-1}^\ell$ holds in Ξ_g. Applying F yields $(\iota_j, \varepsilon_j^\ell) \circ (\iota_i, \varepsilon_i^k)$ and $(\iota_i, \varepsilon_i^k) \circ (\iota_{j-1}, \varepsilon_{j-1}^\ell)$, which coincide since $\iota_j \circ \iota_i = \iota_i \circ \iota_{j-1}$ and the resulting ε-components agree pointwise by inspection.

Group relations, for $\sigma, \tau \in S_n$, functoriality follows immediately from $(\sigma, \mathrm{const}_{\lrcorner}) \circ (\tau, \mathrm{const}_{\lrcorner}) = (\sigma \circ \tau, \mathrm{const}_{\lrcorner})$, and identities are preserved.

Permutation–face interchange. For $\theta \in S_n$ and i, both composites $F(\theta) \circ F(d^k_{\theta^{-1}(i)})$ and $F(d_i^k) \circ F(d_i\theta)$ have first component $\theta \circ \iota_{\theta^{-1}(i)} = \iota_i \circ d_i\theta$. A direct application of the composition rule in Ξ shows that their ε-components coincide, taking value k at i and $\lrcorner$ elsewhere. Thus F preserves all defining relations and induces a well-defined functor $F : \Xi_g \to \Xi$.

Uniqueness. Since Ξ_g is presented by generators and relations, F is uniquely determined by its action on generators. ◀

▶ **Definition 6.** *The* canonical presentation *of a morphism v of Ξ_g is an equation $v = d_{i_r}^{k_r} \circ \cdots \circ d_{i_2}^{k_2} \circ d_{i_1}^{k_1} \circ \tau$ such that $i_1 < \cdots < i_r$.*

▶ **Lemma 7.** *Every morphism v of Ξ_g has a canonical presentation.*

Proof. Fix any presentation $v = u_1 \circ \cdots \circ u_m$ over the generators $\{d_i^k\} \cup \{\theta \in \mathfrak{S}_*\}$. We construct, by induction on m, a factorisation $\mathrm{NF}(v) = D \circ \tau$ with $D = d_{i_r}^{k_r} \circ \cdots \circ d_{i_1}^{k_1}$, $i_1 < \cdots < i_r$, $\tau \in \mathfrak{S}_*$, and show that $\mathrm{NF}(v)$ equals v in Ξ_g (i.e. is obtained by applying only the defining relations).

Base case $m = 0$. $\mathrm{NF}(\mathrm{id}) = \mathrm{id} \circ \mathrm{id}$ is canonical.

Inductive step. Write $v = w \circ g$ with g a generator and suppose $\mathrm{NF}(w) = D \circ \sigma$ is canonical, where $D = d_{i_s}^{\ell_s} \circ \cdots \circ d_{i_1}^{\ell_1}$ with $\sigma \in \mathfrak{S}_*$. We define $\mathrm{NF}(v)$ by cases on g.

Case 1: $g = \theta \in \mathfrak{S}_*$ **(a permutation).** Put $\mathrm{NF}(w \circ \theta) := D \circ (\theta\,\sigma)$. This uses only the group law (Def. 4.3(b)) and clearly preserves canonicity of the coface block.

Case 2: $g = d_i^k$ **(a coface).** First *transport d_i^k through the current permutation σ to the left* using the permutation–face interchange in the form $\sigma \circ d_i^k = d^k_{\sigma(i)} \circ d_{\sigma(i)}\sigma$, which follows from $d_j^k \circ d_j\theta = \theta \circ d^k_{\theta^{-1}(j)}$ by setting $\theta = \sigma$, $j = \sigma(i)$. Hence $D \circ \sigma \circ d_i^k = D \circ d^k_{\sigma(i)} \circ (d_{\sigma(i)}\sigma)$. Set $\sigma' := d_{\sigma(i)}\sigma \in \mathfrak{S}_{*-1}$ (the induced permutation on one fewer coordinate). It remains to *insert* the single coface $d^k_{\sigma(i)}$ into the sorted block D.

<u>Insertion lemma.</u> Given a sorted block $D = d_{i_s}^{\ell_s} \circ \cdots \circ d_{i_1}^{\ell_1}$ with $i_1 < \cdots < i_s$ and a coface d_j^k on its right, there exists a finite sequence of applications of the cubical identity

$$d_q^\lambda \circ d_p^\kappa \;=\; d_p^\kappa \circ d_{q-1}^\lambda \qquad (p < q) \tag{$\natural$}$$

that rewrites $D \circ d_j^k$ into a sorted block D' $D \circ d_j^k \;\equiv\; d^{\ell'_{s'}}_{i'_{s'}} \circ \cdots \circ d^{\ell'_1}_{i'_1} \quad i'_1 < \cdots < i'_{s'}$, where each swap with a left neighbour d_p^κ (with $p < j$) replaces the pair by $d_p^\kappa \circ d^k_{j-1}$, i.e. decrements the index of the moving coface by 1. Concretely: starting from the right, compare j with i_s; while $j \le i_t$, apply ($\flat$) to swap with $d^{\ell_t}_{i_t}$ and update $j \leftarrow j-1$; when $j > i_t$, stop and place d^k_j immediately to the left of $d^{\ell_t}_{i_t}$. This terminates since j strictly decreases on each swap.

Applying the insertion lemma with $j = \sigma(i)$ yields a sorted block D' such that $D \circ d^k_{\sigma(i)} \equiv D'$ with indices strictly increasing. Define $\mathrm{NF}(w \circ d^k_i) := D' \circ \sigma'$. By construction we used only the interchange law and the cubical identities, hence $\mathrm{NF}(w \circ d^k_i)$ is equal to $w \circ d^k_i$ in Ξ_g, and its coface block is sorted. In both cases we obtain a canonical form for v, completing the induction. Therefore every morphism v of Ξ_g admits a presentation of the form $d^{k_r}_{i_r} \circ \cdots \circ d^{k_1}_{i_1} \circ \tau$ with $i_1 < \cdots < i_r$. ◀

▶ **Lemma 8.** *F is surjective.*

Proof. Let $(f, \varepsilon) : (m, \lambda) \to (n, \mu)$ be a morphism in Ξ. Set $J := f([m]) \subseteq (n)$ and $I := (n) \setminus J = \{i_1 < \cdots < i_r\}$ $(r = n - m)$. Let $\iota_J : J \hookrightarrow (n)$ be the inclusion and let $\rho : (m) \xrightarrow{\cong} J$ be the unique increasing bijection ($J = \{j_1 < \cdots < j_m\}$, so $\rho(t) = j_t$). Define the permutation $\tau := \rho^{-1} \circ f \in \mathfrak{S}_m$, so that $f = \iota_J \circ \rho \circ \tau$. Then τ is the unique permutation satisfying $\tau(i) < \tau(i')$ iff $f(i) < f(i')$; equivalently, $f \circ \tau^{-1}$ is order-preserving. Define in Ξ_g $\alpha := d^{k_r}_{i_r} \circ \cdots \circ d^{k_1}_{i_1} \circ \tau$ with $k_s := \varepsilon(i_s) \in \{0, 1\}$.

First component. Write $\iota_I := \iota_{i_r} \circ \cdots \circ \iota_{i_1} : (m) \hookrightarrow (n)$. *Claim.* $\iota_I = \iota_J \circ \rho$. (Induction on r: composing the standard injections that skip I yields the order-preserving bijection $(m) \to J$ followed by inclusion.) This is because for each $t \in (m)$, by definition $\iota_I(t)$ is the t-th smallest element of J, i.e. $\iota_I(t) = j_t$. Since ι_J is the inclusion of J, $(\iota_J \circ \rho)(t) = \iota_J(j_t) = j_t = \iota_I(t)$, for all t. Hence $\iota_I = \iota_J \circ \rho$. Hence the first component of $F(\alpha)$ equals

$$\iota_{i_r} \circ \cdots \circ \iota_{i_1} \circ \tau = \iota_I \circ \tau = (\iota_J \circ \rho) \circ \tau = f \quad \text{by } (*).$$

Second component. Using $F(d^{k_s}_{i_s}) = (\iota_{i_s}, \varepsilon^{k_s}_{i_s})$ and the composition rule in Ξ, a trivial induction on r gives after composing the r cofaces

$$\tilde{\varepsilon}(v) = \begin{cases} k_s & \text{if } v = i_s \in I, \\ \lrcorner & \text{if } v \in J. \end{cases}$$

Composing with $F(\tau) = (\tau, \mathrm{const}_{\lrcorner})$ does not change $\tilde{\varepsilon}$ (the status $\lrcorner$ is constant and the injection image is J). Thus the second component of $F(\alpha)$ is $v \mapsto \begin{cases} \varepsilon(i_s) = k_s & \text{if } v = i_s \in I, \\ \lrcorner & \text{if } v \in J = f([m]), \end{cases}$ which is exactly ε. Therefore $F(\alpha) = (f, \varepsilon)$, so F is surjective. ◀

▶ **Lemma 9.** *F is injective.*

Proof. Let α, α' be morphisms with canonical presentations $\alpha = d^{k_r}_{i_r} \circ \cdots \circ d^{k_1}_{i_1} \circ \tau$, and $\alpha' = d^{k'_{r'}}_{i'_{r'}} \circ \cdots \circ d^{k'_1}_{i'_1} \circ \tau'$, where $i_1 < \cdots < i_r$ and $i'_1 < \cdots < i'_{r'}$. Assume $F(\alpha) = F(\alpha') = (f, \varepsilon)$; we show the canonical data coincide. *Step 1 (recover I and k's from ε).* In Ξ we have $\varepsilon^{-1}(\lrcorner) = f([m])$. Hence $I := (n) \setminus f([m]) = \{v \in (n) \mid \varepsilon(v) \neq \lrcorner\}$. But for a canonical word, $I = \{i_1 < \cdots < i_r\}$ and $\varepsilon(i_s) = k_s$. Thus I and the tuple $(k_s)^r_{s=1}$ are uniquely determined by (f, ε). Applying the same to α' yields $I' = \{i'_1 < \cdots < i'_{r'}\} = I$ and $k'_s = k_s$, so $r' = r$, $i'_s = i_s$ and $k'_s = k_s$ for all s.

Step 2 (recover τ from f). Let $J := f([m])$ and $\rho : (m) \xrightarrow{\cong} J$ be the increasing bijection. By the insertion-of-holes identity $\iota_{i_r} \circ \cdots \circ \iota_{i_1} = \iota_J \circ \rho$, the first component of $F(\alpha)$ is $f = (\iota_J \circ \rho) \circ \tau$. Hence $\tau = \rho^{-1} \circ f$, which is uniquely determined by f. The same computation for α' gives $\tau' = \rho^{-1} \circ f = \tau$. We have shown that different canonical data

Table 1 Base categories at a glance.

Category	Objects	Morphisms & generators
$\square$	canonical conclists	conclist maps; generated by coface maps
Ξ	canonical concsets	concset maps; generated by coface maps and permutations

cannot map to the same (f, ε): from $F(\alpha) = F(\alpha')$ we obtained $r = r'$, $i_s = i'_s$, $k_s = k'_s$, and $\tau = \tau'$. Thus α and α' have the same canonical presentation, hence are equal in Ξ_g. Therefore F is injective. ◀

Once surjectivity and injectivity have been established, we obtain the following result.

▶ **Proposition 10.** *The functor $F : \Xi_g \to \Xi$ is an isomorphism of categories.*

Consequently, the subcategories obtained by forgetting permutations are also isomorphic, that is, $\square_g \cong \square$. **From now on, we simply write $\Xi = \Xi_g$ and $\square = \square_g$**, and freely use their descriptions by generators and relations. The base categories that will be used throughout the remainder of the paper are summarized in Table 1.

3 Precubical sets and HDA

We now turn to the presheaf formulation of HDAs over these bases. We establish that the symmetric HDAs of Kahl [17] (based on Ξ_g) and those of Struth and Ziemiański [22] (based on Ξ) are categorically isomorphic. **Crucially, this equivalence allows us to leverage Kahl's Symmetrization Theorem**, the result that every HDA is hhp-bisimilar to its symmetric expansion. This justifies our order-free semantics as a behavioural model for the **entire class of HDAs, not only the symmetric ones**.

3.1 Precubical sets over a base

▶ **Definition 11** (Precubical set over a base)**.** *Let $\mathbb{C} \in \{\square, \Xi\}$ be one of the base categories introduced above. A* precubical set over $\mathbb{C}$ *is a presheaf $X : \mathbb{C}^{\mathrm{op}} \to$ **Set**. A* map of precubical sets *$g : X \Rightarrow Y$ over $\mathbb{C}$ is a natural transformation. We write $\widehat{\mathbb{C}}$ for the corresponding presheaf category.*

One can equally define presheaves over the symmetric base Ξ_g. However, since we have shown that Ξ_g and Ξ are isomorphic, these bases give rise to equivalent presheaf categories, as stated below.

▶ **Proposition 12** (Equivalence of presheaf categories)**.** *There is an isomorphism of presheaf categories $\widehat{\Xi} \cong \widehat{\Xi}_g$.*

Proof. Since by Prop 10 $F : \Xi_g \xrightarrow{\cong} \Xi$ is an isomorphism of categories, the induced pullback and pushforward along F yield an isomorphism of presheaf categories $\widehat{\Xi} \cong \widehat{\Xi_g}$. ◀

Terminology and correspondence.

- **Case $\mathbb{C} = \square$.** This yields the classical notion of a *precubical set* in the sense of van Glabbeek [23] and Fahrenberg et al. [10, 11], based on ordered concurrency lists.
- **Case $\mathbb{C} = \Xi$.** This gives rise to the *symmetric precubical sets* of Kahl [17] and of Struth and Ziemiański [22], where concurrent events are represented without an explicit order. Since $\Xi \simeq \Xi_g$, both constructions define the same class of *symmetric precubical sets.*

Notation.

- For any coface morphism $d^k_{i,n} : U \to V$, with $|U| = n$, in the chosen base $\mathbb{C} \in \{\square, \Xi\}$ we set $\delta^k_{i,n} := X[d^k_{i,n}]$, and this notation is used uniformly for all bases.
- Given an n-cell x and a permutation $\tau \in \mathfrak{S}_n$, we write $\tau \cdot x$ for $X[\tau](x) \in X(n, \lambda \circ \tau^{-1})$

If X is a presheaf over a category $\mathbb{C}$, where $\mathbb{C}$ is either $\square$ or Ξ, and U is an object $X[U]$ represents the set of those n-cells in X in which exactly the events in U are active. Elements of $X[U]$, form the set of cells Cell_X of X. Specifically, we have:

$$\mathrm{Cell}_X = \bigsqcup_{U \in \mathrm{obj}(\mathbb{C})} X[U].$$

For any $x \in X[U]$, elements of U are called events of x. We write $\mathsf{ev}(x) = U$.

$$\begin{tikzcd}
c \arrow[r, "t"] & d \arrow[r, "u"] & e \\
a \arrow[u, "q"] \arrow[r, "p"'] \arrow[ur, phantom, "x"] & b \arrow[u, "s"'] \arrow[r, "v"'] \arrow[ur, phantom, "y"] & f \arrow[u, "l"']
\end{tikzcd}$$

Figure 3 Precubical set X with two 2-dimensional cells.

▶ **Example 13** (Example of a precubical set)**.** Figure 3 shows a visualization of a precubical set X, in which all one dimensional cells have the same label. So we ignore λ in the conclist notation and write (n). The precubical set X is given by the following:

$$X[0] = \{a, b, c, d, e, f\}, \quad X[1] = \{p, q, s, t, u, v, l\}, \quad X[2] = \{x, y\},$$

and face maps:

$$\begin{aligned}
&\delta^0_1(p) = a,\ \delta^1_1(p) = b, \quad \delta^0_1(q) = a,\ \delta^1_1(q) = c, \quad \delta^0_1(u) = d,\ \delta^1_1(u) = e,\\
&\delta^0_1(s) = b,\ \delta^1_1(s) = d, \quad \delta^0_1(t) = c,\ \delta^1_1(t) = d, \quad \delta^0_1(v) = b,\ \delta^1_1(v) = f,\\
&\delta^0_2(x) = q,\ \delta^1_2(x) = s, \quad \delta^0_1(x) = p,\ \delta^1_1(x) = t,\\
&\delta^0_2(y) = s,\ \delta^1_2(y) = l, \quad \delta^0_1(y) = v,\ \delta^1_1(y) = u.
\end{aligned}$$

In the figure, Σ has one element and the indices refer to the names of the cells.

▶ **Example 14.** Let $Y \in \widehat{\square}$ be the precubical set with a single 2–cell $x \in Y[(2, ab)]$, and no cells in higher dimensions. Such a configuration cannot occur in a symmetric precubical set. Indeed, the transposition $\tau : (2, ab) \xrightarrow{\cong} (2, ba)$ is a morphism in Ξ, so for any $Z \in \widehat{\Xi}$, the induced map $Z[\tau] : Z(2, ab) \to Z(2, ba)$ is a bijection. Hence, the existence of a cell in $Z(2, ab)$ forces the existence of a corresponding cell in $Z(2, ba)$. This obstruction persists in the autoconcurrent case $a = b$. If x were the only 2–cell, symmetry would imply $\tau \cdot x = x$ for the transposition $\tau = (2\ 1)$, yielding $d^0_1 x = d^0_1(\tau \cdot x) = d^0_{\tau^{-1}(1)} x = d^0_2 x$, a contradiction.

▶ **Definition 15** (Higher-dimensional automaton over a base)**.** *Let $\mathbb{C} \in \{\square, \Xi\}$. An* HDA *over $\mathbb{C}$ is a pair $\mathcal{X} = (X, i_X)$ where:*

- *$X \in \widehat{\mathbb{C}}$ is a precubical set over $\mathbb{C}$;*
- *$i_X \in X[0]$ is the initial 0–cell.*

For a natural transformation $f : X \Rightarrow Y$, we write $f_U : X[U] \to Y[U]$ for its component at $U \in \mathbb{C}$. Given HDAs $\mathcal{X} = (X, i_X)$ and $\mathcal{Y} = (Y, i_Y)$ over $\mathbb{C}$, a morphism *$\mathcal{X} \to \mathcal{Y}$ is a natural transformation $f : X \Rightarrow Y$ such that its 0-component preserves the chosen initial cell, i.e. $f_0(i_X) = i_Y$.*

HDAs over $\square$ coincide with the classical (ordered) HDAs, while HDAs over Ξ are the symmetric HDAs.

Final cells are omitted here without loss of generality: HDAs need not have designated final states; these can be added as an external structure when modelling successful termination, but are irrelevant for the properties studied here.

▶ **Definition 16** (Symmetriser; [17]). *Let $X \in \widehat{\square}$ be a precubical set. The* free symmetric precubical set *generated by X is the presheaf $SX \in \widehat{\Xi}$ defined as follows:*

- On objects. *For each canonical conclist (n, λ),*

$$(SX)(n, \lambda) \;=\; \{\, (\theta, x) \mid \theta \in \mathfrak{S}_n,\; x \in X(n, \lambda \circ \theta) \,\}.$$

- On cofaces. *For $d_i^k : (n-1, \lambda \circ \iota_i) \to (n, \lambda)$,*

$$(SX)[d_i^k](\theta, x) \;=\; \big(d_i\theta,\; X[d^k_{\theta^{-1}(i)}](x)\big),$$

where $d_i\theta \in \mathfrak{S}_{n-1}$ is the induced permutation on faces defined in ((1)).

- On permutations. *For $\tau \in \mathfrak{S}_n$,*

$$(SX)[\tau](\theta, x) \;=\; (\tau\theta,\; x) \;:\; (SX)(n, \lambda) \longrightarrow (SX)(n, \lambda \circ \tau^{-1}).$$

The above assignments respect the cubical identities and the permutation–face interchange law of Ξ, so they extend uniquely to a functor $SX : \Xi^{\mathrm{op}} \to \mathbf{Set}$. We call SX the symmetrisation *of X. For an HDA $\mathcal{X} = (X, i_X)$ over $\square$, the* free symmetric HDA *is*

$$S\mathcal{X} := (SX,\; (\mathrm{id}.i_X))$$

For each n–cell $x \in X[n, \lambda]$ and $\tau \in \mathfrak{S}_n$, the element $(\theta, x) \in (SX)(n, \lambda \circ \theta^{-1})$ represents the same n concurrent events as x, but with their linear order permuted by θ. In the symmetric interpretation, this element corresponds to the cell $\tau.x$. Hence, the symmetrisation SX contains all cells of X together with their symmetric variants – one for each permutation of their n events. Consequently, every n–cell of X gives rise to $n!$ distinct copies in SX.

Terminology. Since our interest lies in symmetrisations, we use the following terminology. For a precubical set X, we call SX its *symmetrised precubical set*; objects of this form will be referred to as *sprecubical sets.* For an HDA $\mathcal{X} = (X, i_X)$, we write $\mathcal{X} := (SX,\, (\mathrm{id}.i_X))$ for its *symmetric HDA* (or *sHDA*).

Remark. There also exists a functor $s^* : \widehat{\Xi} \to \widehat{\square}$ obtained by precomposition with the inclusion $s : \square \hookrightarrow \Xi$, that is, $s^*(X) := X \circ s$. This functor keeps the same underlying sets of cells and the same coface maps, but discards all symmetry morphisms of Ξ. Consequently, it also forgets the identifications between a cell and its permuted copies that were induced by those symmetries in the first place. For a symmetric precubical set Y, the ordered precubical set s^*Y thus contains all cubes of Y, but treats previously equivalent permuted cells as independent ones. For instance, if $Y = SX$ as in Example 14, then Y contains a 2–cell in shape $(2, ab)$ together with its permuted companion in shape $(2, ba)$. After applying s^*, both

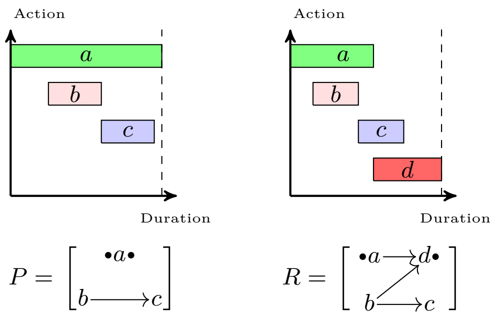

Figure 4 Interval ipomsets (below) with their corresponding interval representations (above). An event with a dot on the left (resp. on the right) is an element of a source (resp. target) interface. Full arrows indicate precedence order.

cells remain present, but the symmetry morphism between them is no longer available; in s^*Y they are interpreted as two independent ordered squares. Applying the symmetriser again to s^*Y freely adds new symmetric copies of each of these cells, so y and $(12).y$ each generate their own pair of permuted variants. In general, every n–cell of Y gives rise to $n!$ copies after the first symmetrisation, $(n!)!$ copies after the second, and so on, resulting in a factorial growth under iteration.

4 Pomsets

Partially ordered multisets (*pomsets*) have long been used to model true concurrency and have played a central role in theoretical and applied concurrency research [13, 14, 20]. Earlier work on higher-dimensional automata (HDAs) extended pomsets by adding an event order on incomparable events and by using interfaces to distinguish the beginning and end of a run [9, 10]. In this paper we drop the event order: unlike [10], we describe the observable content of HDA paths using pomsets that carry only the precedence order, while following the interface construction and gluing composition of [9].

▶ **Definition 17** (Pomset). *A* partially ordered multiset (pomset) *is a tuple* $(P, <_P, \lambda_P)$ *where* P *is a finite set,* $\lambda_P : P \to \Sigma$ *is a labelling function over an alphabet* Σ*,* $<_P$ *is a strict partial order on* P *called* precedence order*.*

To model concurrency using pomsets, elements of a pomset P represent *events*, and $x <_P y$ signifies that event x must occur before event y. Two distinct events $x, y \in P$ are *concurrent*, written $x \parallel y$, precisely when neither $x <_P y$ nor $y <_P x$. An event $x \in P$ is *minimal* if no event precedes it and *maximal* if it precedes no event; we denote the sets of minimal and maximal events by $P_{\min}$ and $P_{\max}$, respectively. A subset $Q \subseteq P$ is an *antichain* if any two distinct elements of Q are concurrent. An antichain is *maximal* if it is not properly contained in any larger antichain. In a pomset, antichains correspond precisely to concsets.

▶ **Definition 18** (ipomset). *Let* $U = (m, \lambda)$ *and* $V = (l, \gamma)$ *be canonical concsets. A* pomset with interfaces (ipomset) *consists of a pomset* P *and two* injective, label-preserving *functions* $U \xrightarrow{s_P} P \xleftarrow{t_P} V$*, such that* $s_P(U) \subseteq P_{\min}$*, called the source interface, and* $t_P(V) \subseteq P_{\max}$*, called the target interface. We denote it by* $\mathbf{P} = (s_P,\ P,\ t_P) :\ U \to V$*.*

The source and target interfaces of an ipomset are antichains, hence concsets. In this paper we work exclusively with interfaces whose domains are *canonical concsets* and whose maps are canonical, namely either the identity or the standard injection $\iota_i \colon (n{-}1) \hookrightarrow (n)$. Although concsets in general admit multiple label-preserving bijections (see Example 20), all interfaces appearing here are uniquely determined, and no further choices arise.

As we drop the event order, we follow the gluing construction for pomsets with interfaces from [9]. This gluing operation is associative, admits a unit, but is not commutative; commutativity is neither expected nor needed here, since gluing models sequential concatenation of paths.

▶ **Definition 19** (Gluing composition of ipomsets). *Let* $(s_1, P_1, t_1) : (n, \nu) \to (m, \mu)$ *and* $(s_2, P_2, t_2) \colon (m, \mu) \to (k, \kappa)$ *be ipomsets. Their* gluing composition *is the ipomset* $(s_1,\ P_1 * P_2,\ t_2) \ :\ (n, \nu) \ \longrightarrow\ (k, \kappa)$, *where the carrier is* $P_1 * P_2 := \big(\ (P_1 \sqcup P_2)\,/\, t_1(i) = s_2(i),\ \leq, \lambda_1 \cup \lambda_2\big)$, *the disjoint union of* P_1 *and* P_2 *with the interface elements identified:* $t_1(i) \equiv s_2(i)$ *for all* $i \in [m]$, *the precedence order is* $\leq\, = \big(\leq_1\ \cup\ \leq_2\ \cup (P_1 \backslash t_1[m]) \times (P_2 \backslash s_2[m])\big)$ *and the labelling function is inherited component-wise:* λ_1 *on* P_1 *and* λ_2 *on* P_2. *We regard* P_1 *and* P_2 *as sub-pomsets of* $P_1 * P_2$ *via the canonical injections.*

Although different choices of interface identifications may in general lead to non-isomorphic gluing results (see Example 20), such situations will not arise in this paper. Indeed, all ipomsets we consider have interfaces whose domain and codomain are canonical concsets, so the gluing operation is uniquely determined. Moreover, every interface map f is strictly order preserving in the sense that for all i, j, $i <_{\mathbb{N}} j \iff f(i) <_{\mathbb{N}} f(j)$, where $<_{\mathbb{N}}$ denotes the usual strict order on natural numbers. In particular, interface identifications preserve $<_{\mathbb{N}}$, ruling out any ambiguity in the gluing construction.

▶ **Example 20.** Changing the identification of interface events may lead to non-isomorphic gluing results. For instance, consider the following two gluings of discrete ipomsets.
Order-preserving identification. Identifying the interfaces via the order-preserving bijection $t_1(i) = s_2(i)$ for $i = 1, 2$ yields

$$\begin{pmatrix} a \longrightarrow b = t_1(1) \bullet \\ \\ \hfill b = t_1(2) \bullet \end{pmatrix} \ * \ \begin{pmatrix} \bullet b = s_2(1) \longrightarrow c \\ \\ \bullet b = s_2(2) \hfill \end{pmatrix} \ = \ \begin{pmatrix} a \longrightarrow b \longrightarrow c \\ b \end{pmatrix}\ .$$

This is the situation that arises throughout this paper.
Order-reversing identification. If instead the interface is identified via the permutation $t_1(1) = s_2(2)$ and $t_1(2) = s_2(1)$, one obtains

$$\begin{pmatrix} a \longrightarrow b = t_1(1) \bullet \\ \\ \hfill b = t_1(2) \bullet \end{pmatrix} \ * \ \begin{pmatrix} \bullet b = s_2(2) \longrightarrow c \\ \\ \bullet b = s_2(1) \hfill \end{pmatrix} \ = \ \left(\begin{tikzcd} a \arrow[r] \arrow[dr] & b \\ b \arrow[r] & c \end{tikzcd}\right)\ .$$

Such a gluing is admissible in general for ipomsets, but it violates order preservation of the interfaces and therefore does not occur under the assumptions imposed in this paper.

▶ **Definition 21** (Interval Ipomset [12]). *An* interval ipomset *is an ipomset* P *such that for any* $x, y, z, w \in P$, *if* $x <_P z$ *and* $y <_P w$, *then* $x <_P w$ *or* $y <_P z$. *In other words, it does not contain an induced subpomset of the form:* $2{+}2 = \begin{bmatrix} \bullet \longrightarrow \bullet \\ \bullet \longrightarrow \bullet \end{bmatrix}$.

▶ **Proposition 22** ([10, 15]). *Let* P *be an ipomset. Then* P *is an interval ipomset if and only if* P *can be expressed as a finite gluing of discrete ipomsets.*

In this work, we focus solely on interval ipomsets: **all ipomsets are assumed to be interval even if not stated explicitly.**

▶ **Definition 23** (Isomorphism of ipomsets). *Let* $\mathbf{P} = (s_P, P, t_P)\colon U \to V$ *and* $\mathbf{Q} = (s_Q, Q, t_Q)\colon U_Q \to V_Q$ *be ipomsets. An* isomorphism $\mathbf{P} \cong \mathbf{Q}$ *is a bijection* $f\colon P \to Q$ *such that* $\lambda_P = \lambda_Q \circ f, x <_P y \iff f(x) <_Q f(y)$, *and* f *restricts to bijections* $f_U : U \to U_Q$ *and* $f_V : V \to V_Q$ *satisfying* $s_Q \circ f_U = f \circ s_P$ *and* $t_Q \circ f_V = f \circ t_P$. *Equivalently, the following diagram commutes:*

$$\begin{tikzcd}
U \arrow[rr, "s_P"] \arrow[d, "f_U"'] & & P \arrow[d, "f"] & & V \arrow[ll, "t_P"'] \arrow[d, "f_V"] \\
U_Q \arrow[rr, "s_Q"'] & & Q & & V_Q \arrow[ll, "t_Q"]
\end{tikzcd}$$

There is at most one isomorphism between two ipomsets in the presence of event order [10]. This uniqueness fails in our case.

Remark (Numbering vs. ordering of interfaces). The integers $\{1, \ldots, n\}$ appearing in a canonical concset (n, λ) are names for interface slots, not a ranking of the concurrent events inside the pomset carrier. The order-preservation constraint imposed on interface maps in Definition 18 is a canonicality condition: it ensures that the gluing operation of Definition 19 is uniquely determined, ruling out the ambiguity illustrated in Example 20. It does not impose any order on the carrier itself. The precedence order $<_P$ on P remains a partial order, and concurrent events stay incomparable. Accordingly, the notion of isomorphism in Definition 23 admits any label-preserving bijection on the carrier that respects precedence and interfaces, with no requirement to preserve event order. This is precisely what allows $(a \parallel b)$ and $(b \parallel a)$ to be identified in our setting, whereas ordered approaches cannot identify them because their isomorphisms must additionally preserve the event order on concurrent events.

5 Paths in HDAs and sHDAs

We recall the notion of paths in HDAs and sHDAs and the standard relations between them, fixing notation for later use. We also make explicit the relationship between paths and their symmetric liftings.

▶ **Definition 24.** *A* path *of length* n *in a precubical set* X *is a sequence* $\alpha = (x_0, \varphi_1, x_1, \varphi_2, \ldots, \varphi_n, x_n)$, *where* $x_j \in X[U_j]$ *are cells, and for all* j, *either*

- $\varphi_j = d^0_{i_j} \in \square(U_{j-1}, U_j)$ *a source map and* $x_{j-1} = \delta^0_{i_j}(x_j)$ *(up-step), or*
- $\varphi_j = d^1_{i_j} \in \square(U_j, U_{j-1}), \delta^1_{i_j}(x_{j-1}) = x_j$ *(down-step).*

A path in a sprecubical set Y *is a path in the underlying precubical set* s^*Y.

▶ **Lemma 25** (Paths in a symmetrised precubical set). *Let* $X \in \widehat{\square}$ *and consider its symmetrisation* $SX \in \widehat{\Xi}$. *A* path *of length* n *in the sprecubical set* SX *is equivalently a sequence* $\alpha = \big((\tau_0.x_0), \varphi_1, (\tau_1.x_1), \ldots, \varphi_n, (\tau_n.x_n)\big)$, *where each* $x_j \in X[U_j]$, *each* $\tau_j \in \mathfrak{S}_{|U_j|}$ *is a permutation, and each* $\varphi_j = d^{k_j}_{p_j}$ *is a coface map in* Ξ, *such that for every* $j = 1, \ldots, n$:

(i) (step condition) *either*
- $k_j = 0$ *and* $(\tau_{j-1}.x_{j-1}) = (SX)[d^0_{p_j}](\tau_j.x_j)$, *or*
- $k_j = 1$ *and* $(\tau_j.x_j) = (SX)[d^1_{p_j}](\tau_{j-1}.x_{j-1})$.

(ii) (permutation coherence) *the permutations satisfy*

$$\tau_{j-1} = d_{p_j}\tau_j \text{ if } k_j = 0, \qquad \text{and} \qquad \tau_j = d_{p_j}\tau_{j-1} \text{ if } k_j = 1.$$

Proof. By definition, a path in the sprecubical set $SX \in \widehat{\Xi}$ means a path in the underlying precubical set $s^*(SX) \in \widehat{\square}$, i.e. a sequence of cells related by face maps in the sense of Definition 24. Unfolding the presheaf action of SX on a coface map d_p^k, we have

$$(SX)[d_p^k](\tau.x) = \big(d_p\tau.\ X[d^k_{\tau^{-1}(p)}](x)\big),$$

where $d_p\tau$ denotes the induced permutation on the corresponding face. Hence, for an up-step ($k = 0$) the condition $(\tau_-.x_-) = (SX)[d_p^0](\tau_+.x_+)$ is equivalent to the pair of equalities $\tau_- = d_p\tau_+$ and $x_- = X[d^0_{\tau_+^{-1}(p)}](x_+)$, and for a down-step ($k = 1$) the condition $(\tau_+.x_+) = (SX)[d_p^1](\tau_-.x_-)$ is equivalent to $\tau_+ = d_p\tau_-$ and $x_+ = X[d^1_{\tau_-^{-1}(p)}](x_-)$. Applying these equivalences stepwise along the sequence yields exactly the up-step/down-step clauses in (i), and the permutation coherence equations in (ii). Conversely, if the equalities in (i) (equivalently, the expanded conditions above) hold at every step, then each consecutive triple forms a valid face step in $s^*(SX)$, hence the whole sequence is a path in SX. ◀

Relation between paths in X and in SX. Let X be a precubical set and let $\alpha = (x_0, d^{k_1}_{i_1}, x_1, \ldots, d^{k_m}_{i_m}, x_m) \in \mathrm{Path}_X$ be a path in X. A *lifting* of α to the sprecubical set SX is a path of the form $\beta = \big((\tau_0.x_0),\ d^{k_1}_{p_1},\ (\tau_1.x_1),\ \ldots,\ d^{k_m}_{p_m},\ (\tau_m.x_m)\big) \in \mathrm{Path}_{SX}$, where the permutations $\tau_0, \ldots, \tau_m$ and indices $p_1, \ldots, p_m$ satisfy

$$p_j = \tau_{j-k_j}(i_j) \quad \text{and} \quad \begin{cases} \tau_{j-1} = d_{p_j}\tau_j & \text{if } k_j = 0, \\ \tau_j = d_{p_j}\tau_{j-1} & \text{if } k_j = 1, \end{cases} \forall j = 1, \ldots, m.$$

Among all liftings of α, the *canonical lifting* $S\alpha = \big((\mathrm{id}.x_0),\ d^{k_1}_{i_1},\ (\mathrm{id}.x_1),\ \ldots,\ d^{k_m}_{i_m},\ (\mathrm{id}.x_m)\big)$ is obtained by choosing $\tau_j = \mathrm{id}$ for all j.

Conversely, Lemma 25 shows that every path $\beta \in \mathrm{Path}_{SX}$ is of this form for a unique underlying path $\alpha \in \mathrm{Path}_X$, obtained by setting $i_j = \tau^{-1}_{j-k_j}(p_j)$.

Path notation and operations. Let $\alpha = (x_0, \varphi_1, \ldots, \varphi_n, x_n)$ be a path in a precubical set X or in a sprecubical set SX. A *path in an HDA* $\mathcal{X} = (X, i_X)$ is a path in the underlying precubical set X whose first cell is the initial cell i_X. Similarly, a path in the symmetric HDA $S\mathcal{X}$ is a path in SX whose first cell is $(\mathrm{id}.i_X)$. We write Path_X and Path_{SX} for the sets of all paths in X and SX, respectively, and $\mathrm{Path}_{\mathcal{X}} \subseteq \mathrm{Path}_X$ and $\mathrm{Path}_{S\mathcal{X}} \subseteq \mathrm{Path}_{SX}$ for the corresponding sets of paths starting at the initial cell. This distinction matters for the semantics below. The ipomset label are defined for *all* combinatorial paths in Path_X and Path_{SX}, independently of the initial cell. By contrast, the ST–trace is defined only for executions, that is, for paths in $\mathrm{Path}_{\mathcal{X}}$ and $\mathrm{Path}_{S\mathcal{X}}$.

If $\alpha = (x_0, \varphi_1, \ldots, \varphi_n, x_n)$ and $\beta = (y_0, \psi_1, \ldots, \psi_m, y_m)$ are paths in X or in SX with $x_n = y_0$, their *concatenation* is the path $\alpha * \beta = \big(x_0, \varphi_1, \ldots, \varphi_n, x_n, \psi_1, y_1, \ldots, \psi_m, y_m\big)$. In this case, we say that α is a *restriction* of $\alpha * \beta$.

▶ **Example 26** (Paths and liftings in the symmetrised square)**.** Consider the HDA $\mathcal{X} = (X, i_X)$ depicted on the left of Figure 5, consisting of a single 2–cell $x \in X(2, ab)$, together with its faces. The symmetrised HDA $S\mathcal{X}$ is shown on the right of the figure and contains two 2–cells, $x \in (SX)(2, ab)$ and $x' \in (SX)(2, ba)$, with $\delta_1^k(x) = \delta_2^k(x')$ for $k = 0, 1$. Consider the path in X $\alpha = \big(\delta_1^0(x),\ d_2^0,\ x,\ d_1^1,\ \delta_2^1(x)\big)$, which enters the square x along the b–edge and exits it along the a–edge. In the symmetrised precubical set SX, the path α admits exactly two liftings. The canonical lifting passes through the cell $x \in (SX)([2, ab])$ and is given by

$S\alpha = \big(\delta_1^0(x),\ d_2^0,\ x,\ d_1^1,\ \delta_2^1(x)\big)$. The second lifting passes through the symmetric copy $x' \in (SX)(2, ba)$. Since the event order is reversed, the face indices are exchanged, yielding the path $\alpha' = \big(\delta_2^0(x'),\ d_1^0,\ x',\ d_2^1,\ \delta_1^1(x')\big)$.

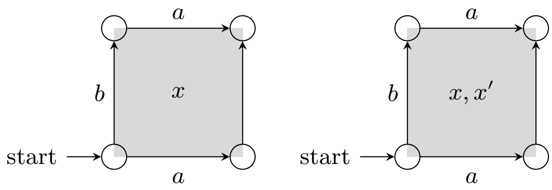

Figure 5 HDA $\mathcal{X}$ on the left with its symmetriser $S\mathcal{X}$ on the right.

The following is an example in 3 dimensions. It illustrates how the symmetriser affects paths and how permutations must act coherently along a path in SX.

▶ **Example 27.** Consider the precubical set X represented in Figure 6, where $x, z \in X[3]$ the two 3-cells sharing the 2-cell y, and consider $\alpha = (x,\ d_3^1,\ y,\ d_3^0,\ z) \in \mathrm{Path}_X$, which first terminates the event c and then initiates the event d. There are $3! = 6$ distinct liftings of α in SX. These include:

1. $\beta = \big((12)\cdot x,\ d_3^1,\ (12)\cdot y,\ d_3^0,\ (12)\cdot z\big)$, the symmetric lifting corresponding to the transposition (12) acting on $\mathsf{ev}(x)$.
2. $\beta' = \big(\sigma\cdot x,\ d_1^1,\ (12)\cdot y,\ d_1^0,\ \sigma\cdot z\big)$, where $\sigma = (123)$, the reduced permutation is $d_1\sigma = (12)$. Here the intermediate 2-cell is $(d_1\sigma)\cdot y = (12)\cdot y$, and the face maps d_1^1 and d_1^0 correspond to d_3^1 and d_3^0 on the underlying path α, as enforced by the equivariance squares (via the permutation σ and the equation $\theta_- = d_i\theta_+$).

We recall the notion of adjacency of paths, as introduced by van Glabbeek [23]. Adjacency and the derived notion of congruence apply uniformly to paths in a precubical set X and in its symmetric counterpart SX [17].

▶ **Definition 28** (Adjacency of paths)**.** *Two paths α and α' are said to be* adjacent*, written $\alpha \leftrightsquigarrow \alpha'$, if one can be obtained from the other by a single local replacement of one of the following forms, for indices $i < j$:*

(1) $(d_i^0,\ x_\ell,\ d_j^0) \rightsquigarrow (d_{j-1}^0,\ x'_\ell,\ d_i^0)$, **(3)** $(d_i^0,\ x_\ell,\ d_j^1) \rightsquigarrow (d_{j-1}^1,\ x'_\ell,\ d_i^0)$,
(2) $(d_j^1,\ x_\ell,\ d_i^1) \rightsquigarrow (d_i^1,\ x'_\ell,\ d_{j-1}^1)$, **(4)** $(d_j^0,\ x_\ell,\ d_i^1) \rightsquigarrow (d_i^1,\ x'_\ell,\ d_{j-1}^0)$.

For any α of length m and $1 \le \ell < m$, there exists a unique path $\alpha^{(\ell)}$ obtained from α by applying the corresponding adjacency replacement to the segment $(\varphi_\ell, x_\ell, \varphi_{\ell+1})$ of α [23]. We write $\alpha \overset{\ell}{\leftrightsquigarrow} \alpha^{(\ell)}$.

Adjacency rules (1) and (2) exchange two faces of the same polarity and are reversible, whereas rules (3) and (4) interchange a start with a termination and are directed. The reversible rules generate a symmetric equivalence relation on paths, called *congruence*, which is the notion relevant for our purposes. The directed rules induce subsumption; see [11].

▶ **Definition 29** (Congruence of paths)**.** *Congruence $\simeq$ is the relation on paths generated by, for indices $i < j$:*

- $(d_i^0,\ x_\ell,\ d_j^0) \simeq (d_{j-1}^0,\ x'_\ell,\ d_i^0)$;
- $(d_j^1,\ x_\ell,\ d_i^1) \simeq (d_i^1,\ x'_\ell,\ d_{j-1}^1)$;
- $\gamma * \alpha * \delta \simeq \gamma * \beta * \delta$ *whenever* $\alpha \simeq \beta$.

where in each case x'_ℓ is the unique cell determined by the adjacency replacement in Definition 28.

6 Observable content

This section collects the notions used to describe the observable behaviour of paths. We recall the ST–trace semantics and introduce an event order free pomset-based notion of path labels. This formulation and its extension to sHDAs are new. We then study how observable content behaves under symmetric liftings.

6.1 ST-trace

Fix a down–step $(x_\ell, d^1_{i_{\ell+1}}, x_{\ell+1})$ occurring in a path

$$\alpha = (x_0, \varphi_1, x_1, \dots, \varphi_\ell, x_\ell, d^1_{i_{\ell+1}}, x_{\ell+1}, \varphi_{\ell+2}, \dots, x_m)$$

in a (s)HDA. By [23, 17], there exists a unique index $k \leq \ell$ such that

$$\alpha \overset{\ell}{\leftrightsquigarrow} \alpha^{(\ell)} \overset{\ell-1}{\leftrightsquigarrow} \alpha^{(\ell-1)} \overset{\ell-2}{\leftrightsquigarrow} \cdots \overset{k+1}{\leftrightsquigarrow} \alpha^{(k+1)} \overset{k}{\not\leftrightsquigarrow} \alpha^{(k)},$$

where each step is obtained by applying the unique adjacency replacement at the indicated position.

This means that the down–step can be moved successively towards the beginning of the path by adjacency replacements in the sense of Definition 28, until it sits immediately after its matching start step and cannot be moved further.

More explicitly, the down–step $d^1_{i_{\ell+1}}$ can be commuted one position to the left precisely when the adjacent segment $(\varphi_\ell, x_\ell, d^1_{i_{\ell+1}})$ matches one of the adjacency patterns of Definition 28, that is, when the indices of the two consecutive face maps satisfy the corresponding inequality condition $i < j$ required there.

The process stops exactly at position k because this condition fails: in $\alpha^{(k+1)}$ the adjacent segment has the form $(d^0_{i_k}, x_{k+1}, d^1_{i_k})$, where the indices coincide and no adjacency replacement is applicable. We then write $\mathsf{start}(i_{\ell+1}) = k$.

▶ **Definition 30** (ST–trace [23]). *Let $\alpha = (x_0, d^{k_1}_{i_1}, x_1, \dots, d^{k_n}_{i_n}, x_n) \in \mathrm{Path}_{S\mathcal{X}}$, and let $\lambda(i_j)$ denote the label of the event whose start or termination is represented by the face map $d^{k_j}_{i_j}$. For each step define*

$$\sigma^{ST}_j := \begin{cases} + & \text{if } k_j = 0, \\ start(i_j) & \text{if } k_j = 1. \end{cases}$$

Then $ST\text{–}trace(\alpha) = (\lambda(i_1)^{\sigma^{ST}_1}, \dots, \lambda(i_n)^{\sigma^{ST}_n})$.

The *ST–trace* records not only when each action starts $(+)$ and terminates, but also links each termination to the position of its corresponding start via the index $start(i_j)$. In this way, the ST–trace encodes the causal pairing between the beginning and ending of individual actions, capturing their overlap and nesting (see [23] for the original formulation).

6.2 Observable content as pomsets

Recall that for a cell x in a (s)precubical set, $\mathsf{ev}(x)$ is a canonical conclist (concset). In this section, we systematically forget the event order and regard $\mathsf{ev}(x)$ only as its underlying concset. We now extend ev from cells to paths.

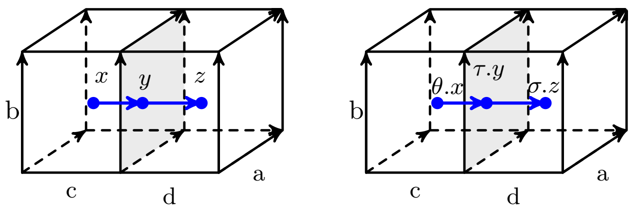

Figure 6 On the left hand side, a precubical set X consisting of two 3-dimensional cubes $x \in X(3, abc)$ and $z \in X(3, abd)$ attached along a common 2-dimensional cell $y \in X(2, ab)$. On the right hand side, Its symmetrization SX, where each cube generates six 3-cells of the form $\theta \cdot x$ and $\sigma \cdot z$ for some $\theta, \sigma \in \mathfrak{S}_3$, all sharing the same boundary faces $\tau.y$ and id.y, where $\tau \in \mathfrak{S}_2$.

▶ **Definition 31** (label of a path). *Let X be a precubical set and let $\alpha \in \mathrm{Path}_X$. The label of α is the ipomset $\mathsf{ev}(\alpha)$, computed recursively*

1. *If $\alpha = (x)$ has length 0, then we set $\mathsf{ev}(\alpha) = (\mathrm{id}_{\mathsf{ev}(x)}, \mathsf{ev}(x), \mathrm{id}_{\mathsf{ev}(x)}) : \mathsf{ev}(x) \to \mathsf{ev}(x)$;*
2. *If $\alpha = \left(y, d_i^0, x\right)$, then $\mathsf{ev}(\alpha)$ is $(\iota_i, \mathsf{ev}(x), \mathrm{id}_{\mathsf{ev}(x)}) : \mathsf{ev}(y) \to \mathsf{ev}(x)$;*
3. *If $\alpha = \left(x, d_i^1, y\right)$, then $\mathsf{ev}(\alpha)$ is $(\mathrm{id}_{\mathsf{ev}(x)}, \mathsf{ev}(x), \iota_i) : \mathsf{ev}(x) \to \mathsf{ev}(y)$;*
4. *If $\alpha = \beta_1 * \cdots * \beta_m$ is a concatenation of steps β_i, then $\mathsf{ev}(\alpha) = \mathsf{ev}\,(\beta_1) * \cdots * \mathsf{ev}\,(\beta_m)$. injection skipping i.*

Since paths in a symmetric precubical set SX are formally paths in the underlying precubical set $s^*(SX)$, the interval–pomset label $\mathsf{ev}(\beta)$ is also well defined for every $\beta \in \mathrm{Path}_{SX}$.

By construction, ipomset labels are built stepwise along paths; in particular, the label of any prefix embeds canonically into the label of the full path.

The label of any path α in a (s)precubical set is a finite gluing of discrete ipomsets. Consequently, by Proposition 22, $\mathsf{ev}(\alpha)$ is an interval ipomset for any path α. This is why we restricted our focus on only interval ipomsets.

Comparison with ordered labels

In the literature, interval–pomset labels of paths are typically defined using conclists, thereby retaining an explicit order on events; this ordering is used to control the gluing operation [10]. In contrast, we systematically forget event order and work with the underlying concsets. The gluing operation remains well defined in our setting because all interfaces involved are canonical, and interface maps are uniquely determined.

Our formulation allows us to identify behaviours that differ only by a permutation of concurrent events. In particular, the interval pomsets $(a \parallel b)$ and $(b \parallel a)$ are isomorphic in our setting, whereas they are distinguished in ordered approaches, where isomorphisms must preserve event order. This resolves a mismatch between pomset semantics and the intended treatment of concurrency that appears in parts of the literature.

▶ **Example 32.** Consider the HDA $\mathcal{X}$ and its sHDA $S\mathcal{X}$ of Figure 6.

- The label of $\alpha = (x, d_3^1, y, d_3^0, z)$ is the ipomset

$$\begin{aligned}\mathsf{ev}(\alpha) &= (\mathrm{id}_{\mathsf{ev}(x)},\, \mathsf{ev}(x),\, \iota_3) * (\iota_3,\, \mathsf{ev}(z),\, \mathrm{id}_{\mathsf{ev}(z)}) : \mathsf{ev}(x) \to \mathsf{ev}(z)\\ &= \begin{pmatrix} \bullet a \bullet \\ \bullet b \bullet \\ \bullet c \end{pmatrix} * \begin{pmatrix} \bullet a \bullet \\ \bullet b \bullet \\ d \bullet \end{pmatrix}\\ &= \begin{pmatrix} & \bullet a \bullet & \\ & \bullet b \bullet & \\ \bullet c & \longrightarrow & d \bullet \end{pmatrix},\end{aligned}$$

which corresponds to first terminating the event c and then starting d through the common face labelled ab.

- The label of $\beta = \big((\theta\cdot x),\ d_3^1,\ (d_3\theta\cdot y),\ d_3^0,\ (\theta\cdot z)\big)$, where $\theta = (12)$, is

$$\mathsf{ev}(\beta) = \begin{pmatrix} \bullet b\bullet \\ \bullet a\bullet \\ \bullet c \end{pmatrix} * \begin{pmatrix} \bullet b\bullet \\ \bullet a\bullet \\ d\bullet \end{pmatrix} = \begin{pmatrix} & \bullet b\bullet & \\ & \bullet a\bullet & \\ \bullet c & \longrightarrow & d\bullet \end{pmatrix}.$$

- The label of $\beta' = \big(\sigma\cdot x,\ d_1^1,\ (12)\cdot y,\ d_1^0,\ \sigma\cdot z\big)$, where $\sigma = (123)$, is

$$\mathsf{ev}(\beta') = \begin{pmatrix} \bullet c \\ \bullet a\bullet \\ \bullet b\bullet \end{pmatrix} * \begin{pmatrix} d\bullet \\ \bullet a\bullet \\ \bullet b\bullet \end{pmatrix} = \begin{pmatrix} \bullet c & \longrightarrow & d\bullet \\ & \bullet a\bullet & \\ & \bullet b\bullet & \end{pmatrix}.$$

Remark on autoconcurrency. All constructions above apply equally in the presence of autoconcurrency. Distinct concurrent events carrying the same label remain distinguished by their structural positions in the underlying concsets. Hence, path formation and interval–pomset labelling are unaffected by autoconcurrency.

6.3 Compatibility of symmetrisation with Observable content

We relate the observable content of a path in a precubical set X to that of its liftings to the symmetric precubical set SX. In the following, we consider separately the behaviour of ST–traces and of interval–pomset labels under symmetrisation.

ST–semantics

Lifting a path to the symmetric setting preserves its ST–trace. In particular, the matching between start and termination events is invariant under symmetrisation. This follows from Theorem 6.1 of [17], which shows that every HDA is hhp-bisimilar to its symmetric counterpart; hhp-bisimulation will be introduced in Section 8.

▶ **Proposition 33.** *Let $\mathcal{X} = (X, i_X, F_X)$ be an HDA, and let $\alpha \in \mathrm{Path}_{\mathcal{X}}$. If α' is a lifting of α in $S\mathcal{X}$, then* $\text{ST-trace}(\alpha') = \text{ST-trace}(\alpha)$.

Thus symmetrisation preserves the causal pairing of events: although SX contains multiple symmetric copies of each cell, these permutations affect only the naming of events, never the temporal structure of the execution.

Pomset semantics

A similar compatibility holds for the ipomset interpretation of paths. The label of a symmetric lifting of a path is obtained by permuting the ipomset label of the original path by the symmetric action.

▶ **Proposition 34.** *For any path α in a precubical set X and $\beta \in SX$ lifting of α, we have*

$$\mathsf{ev}(\alpha) \cong \mathsf{ev}(\beta).$$

Proof. We employ induction on m the length of β (the same as the length of α).

- If $\beta := (\theta.x)$ thus $\alpha := (x)$, then

$$\mathsf{ev}(\beta) \ = \ (\mathrm{id}, \mathsf{ev}(\theta.x),\ \mathrm{id}) : \mathsf{ev}(x) \cong \ (\mathrm{id},\ \mathsf{ev}(x),\ \mathrm{id}) = \mathsf{ev}(\alpha).$$

- If $\beta = ((\theta_{j-1}.y),\ d^0_{i_j},\ (\theta_j.x))$ thus $\alpha = (y, d^0_{\theta_j^{-1}(i_j)}, x))$, then $\mathsf{ev}(\beta) \ := \ \big(\iota_{i_j}\ \mathsf{ev}(\theta_j.x),\ \mathrm{id}\big)$. Further $\mathsf{ev}(\alpha) \ := \ \big(\iota_{\theta_j^{-1}(i_j)}, \mathsf{ev}(x),\ \mathrm{id}\ \big) : \mathsf{ev}(y) \to \mathsf{ev}(x) \cong \mathsf{ev}(\beta)$
- In the case of down step, we proceed similarly.
- If $\beta = \beta_1 * \beta_2$, then $\alpha = \alpha_1 * \alpha_2$ such that β_ℓ is a lifting of α_ℓ for $\ell = \{0,1\}$. By induction hypothesis, there exist isomorphisms $f_1 : \mathsf{ev}(\alpha_1) \xrightarrow{\cong} \mathsf{ev}(\beta_1)$ and $f_2 : \mathsf{ev}(\alpha_2) \xrightarrow{\cong} \mathsf{ev}(\beta_2)$. By the equivarience condition in Lemma 25, and Def.31 of ev, f_1 and f_2 coincide on the interface along which the gluing is performed. Hence the union $f := f_1 \cup f_2$ is a well-defined bijection on the carrier of the glued pomset $\mathsf{ev}(\alpha_1) * \mathsf{ev}(\alpha_2)$. Moreover, f acts on the external interfaces. Therefore f is an isomorphism i.e. $\mathsf{ev}(\alpha) \cong \mathsf{ev}(\beta)$. ◀

These results show that symmetrisation is compatible with both observable semantics: ST–traces are preserved, while interval–pomset labels are preserved up to isomorphism.

7 Relation between Pomset Labels and ST–Traces

This section makes precise the relationship between ST–traces and interval–pomset labels. Although both semantics are widely used to describe the behaviour of paths, their exact correspondence depends on subtle structural conditions that are often left implicit. We first show how equality of ST–traces is reflected at the level of pomset labels, and then analyse the converse direction.

▶ **Lemma 35.** *Let $\alpha \in \mathrm{Path}_{\mathcal{X}}$ and $\beta \in \mathrm{Path}_{\mathcal{Y}}$, where $\mathcal{X}$ and $\mathcal{Y}$ are (s)HDAs. If α and β have the same ST-trace, then $\mathsf{ev}(\alpha) \cong \mathsf{ev}(\beta)$.*

Proof. Write $\alpha = (\alpha_0, \dots, \alpha_n)$ and $\beta = (\beta_0, \dots, \beta_n)$ for paths of the same length, and let $\alpha_{\le m}$ and $\beta_{\le m}$ be their prefixes of length m. We prove by induction on m that $\mathsf{ev}(\alpha_{\le m}) \cong \mathsf{ev}(\beta_{\le m})$ as ipomsets (Def. 23). **Base case $m = 0$.** Both prefixes are the initial cell. Hence $\mathsf{ev}(\alpha_{\le 0})$ and $\mathsf{ev}(\beta_{\le 0})$ are the empty ipomset, so the isomorphism is trivial. **Induction step.** Assume $\mathsf{ev}(\alpha_{\le m}) \cong \mathsf{ev}(\beta_{\le m})$ and consider the last steps $\alpha_m \to \alpha_{m+1}$ and $\beta_m \to \beta_{m+1}$. Set $\pi_m := \mathsf{ev}(\alpha_{\le m}) = (s_m, P_m, t_m) : U_m \to V_m$, and $\rho_m := \mathsf{ev}(\beta_{\le m}) = (s'_m, Q_m, t'_m) : U'_m \to V'_m$. By the induction hypothesis, there exists an isomorphism $(f_m, f^U_m, f^V_m) : \pi_m \cong \rho_m$, that is, $f_m : P_m \to Q_m$ is a label-preserving bijection reflecting and preserving precedence, and $f^U_m : U_m \to U'_m$, $f^V_m : V_m \to V'_m$ are the induced interface bijections making the interface squares commute.

$$
\begin{tikzcd}
U_m \arrow[r, "s_m"] \arrow[d, "f^U_m"'] & P_m \arrow[d, "f_m"] & V_m \arrow[l, "t_m"'] \arrow[d, "f^V_m"] \\
U'_m \arrow[r, "s'_m"'] & Q_m & V'_m \arrow[l, "t'_m"]
\end{tikzcd}
$$

Since ST-trace(α) = ST-trace(β), the $(m{+}1)$-st trace symbol for α and β coincides. We distinguish two cases.

Case 1: the common trace symbol is a^+. Then both last steps are *starts* of an a-labelled event. By definition of ev, we have $\mathsf{ev}(\alpha_{\le m+1}) \ = \ \pi_m * E^+_a$, $\mathsf{ev}(\beta_{\le m+1}) \ = \ \rho_m * E^{+\prime}_a$, where E^+_a and $E^{+\prime}_a$ are the elementary ipomsets that add one fresh a-event, glued along the current target interfaces V_m and V'_m, respectively. There is an evident isomorphism $E^+_a \cong E^{+\prime}_a$ whose induced bijection on the gluing interface is precisely $f^V_m : V_m \to V'_m$ (it maps the unique new a-event to the unique new a-event). by the stepwise definition of path labels, the carrier bijection f_m extends to an isomorphism $\mathsf{ev}(\alpha_{\le m+1}) \ \cong \ \mathsf{ev}(\beta_{\le m+1})$.

$$\begin{tikzcd} P_m \arrow[r, "i_P"] \arrow[d, "f_m"'] & P_{m+1} \arrow[d, dashed, "f_{m+1}"] \\ Q_m \arrow[r, "i'_Q"'] & Q_{m+1} \end{tikzcd} \qquad \begin{tikzcd} R^+_m \arrow[r, "i_R"] \arrow[d, "g_m"'] & P_{m+1} \arrow[d, dashed, "f_{m+1}"] \\ R'^+_m \arrow[r, "i'_R"'] & Q_{m+1} \end{tikzcd}$$

Case 2: the common trace symbol is a^-_k. Then both last steps are *terminations* of the k-th currently-open a-event (in the standard ST bookkeeping). Let $e \in V_m$ be the interface element of π_m corresponding to that k-th open a-event, and let $e' \in V'_m$ be the corresponding element of ρ_m. Because the prefixes have the same ST-trace and (f_m, f^U_m, f^V_m) is an ipomset isomorphism, the induced bijection f^V_m respects this bookkeeping, hence $f^V_m(e) = e'$. By definition of ev, we have $\mathsf{ev}(\alpha_{\le m+1}) = \pi_m * E^-_{a,e}$, $\mathsf{ev}(\beta_{\le m+1}) = \rho_m * {E^-_{a,e'}}'$, where $E^-_{a,e}$ (resp. ${E^-_{a,e'}}'$) is the elementary ipomset that terminates the distinguished interface element e (resp. e'), glued along V_m (resp. V'_m). There is an isomorphism $E^-_{a,e} \cong {E^-_{a,e'}}'$ whose induced bijection on the gluing interface is f^V_m and which maps the distinguished element e to e'.

$$\begin{tikzcd} V_m \arrow[r, "s^-"] \arrow[d, "f^V_m"'] & R^-_m \arrow[d, "g_m"] \\ V'_m \arrow[r, "s'^-"'] & R'^-_m \end{tikzcd}$$

By the stepwise definition of path labels, we obtain $\mathsf{ev}(\alpha_{\le m+1}) \cong \mathsf{ev}(\beta_{\le m+1})$. Thus in either case the isomorphism extends from length m to length $m+1$. By induction, $\mathsf{ev}(\alpha) \cong \mathsf{ev}(\beta)$. ◀

One might expect the converse to hold, namely that $\mathsf{ev}(\alpha) \cong \mathsf{ev}(\beta)$ implies equality of ST–traces. This is not true in general. Ipomset semantics abstract away from the precise temporal order of starts and terminations of concurrent events and therefore identify paths that are merely congruent. For instance, the ST–traces $a^+b^+a^1b^2$ and $b^+a^+a^2b^1$ are distinct but induce the same interval ipomset $(a \parallel b)$. Nevertheless, ipomset isomorphism retains enough information to recover temporal behaviour up to congruence. Achieving this requires additional structure, which we develop next.

▶ **Definition 36.** *Let $\alpha = (x_0, d^{k_1}_{i_1}, x_1, d^{k_2}_{i_2}, .., d^{k_m}_{i_m}, x_m)$ and $\beta = (y_0, d^{k_1}_{r_1}, y_1, d^{k_2}_{r_2}, \ldots, d^{k_m}_{r_m}, y_m)$. We say that α and β have matching events if $\mathsf{ev}(x_{j-1}, d^{k_j}_{i_j}, x_j) = \mathsf{ev}(y_{j-1}, d^{k_j}_{r_j}, y_j)$. We write $\alpha \equiv \beta$.*

▶ **Lemma 37.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be (s)HDAs, and let $\alpha \in \mathrm{Path}_\mathcal{X}$ and $\beta \in \mathrm{Path}_\mathcal{Y}$. If $\alpha \equiv \beta$, then ST-trace(α) =ST-trace(β).*

Proof. By definition of $\equiv$, the paths α and β have the same sequence of face maps. That is, we can write $\alpha = (x_0, d^{k_1}_{i_1}, x_1, \ldots, d^{k_{m+1}}_{i_{m+1}}, x_{m+1})$, and $\beta = (y_0, d^{k_1}_{i_1}, y_1, \ldots, d^{k_{m+1}}_{i_{m+1}}, y_{m+1})$, for the same indices (i_j, k_j). Since the ST-trace construction depends only on the sequence of face maps $(d^{k_1}_{i_1}, \ldots, d^{k_{m+1}}_{i_{m+1}})$ and not on the intermediate cells, it follows immediately that ST-trace(α) = ST-trace(β). ◀

The notion of matching events captures exactly the stepwise information needed to determine the ST–trace. However, matching events alone is insufficient to relate arbitrary paths. To realise ipomset equivalence stepwise, we therefore pass to symmetric HDAs, where all permutations of concurrent events are explicit.

▶ **Lemma 38.** *Let X be a precubical set, and let $\alpha \in \mathrm{Path}_X$ and P an ipomset. If $\mathsf{ev}(\alpha) \cong P$, then there exists $\alpha' \in \mathrm{Path}_{SX}$ lifting of α such that $\mathsf{ev}(\alpha') = P$*

Proof. Fix $\alpha' \in \mathrm{Path}_{SX}$ lifting of α. We employ induction on m, the length of α.

- If $\alpha = (x)$ there exists a unique permutation σ such that $\sigma(\mathsf{ev}(x)) = P$. Define $\alpha' := (\sigma.x)$. Hence $\mathsf{ev}(\alpha') = P$, establishing the base case.
- If $\alpha = \big(y,\, d_i^0,\, x\big)$. On the one hand $\mathsf{ev}(\alpha) = \big(\iota_i,\ \mathsf{ev}(x), \mathrm{id}\big) : \mathsf{ev}(y) \to \mathsf{ev}(x)$. Since $\mathsf{ev}(\alpha) \cong P$ by an isomorphism f (defined uniquely by permutation $\theta \in \mathfrak{S}$), we have the following diagram

$$\begin{tikzcd} U_m \arrow[r, "\iota_i"] \arrow[d, "f^U"'] & \mathsf{ev}(x) \arrow[d, "f"] & \mathsf{ev}(x) \arrow[l, "\mathrm{id}"'] \arrow[d, "f"] \\ U \arrow[r, "\iota_{f(i)}"'] & V & V \arrow[l, "\mathrm{id}"] \end{tikzcd}$$

  Define $\alpha' = \big((d_i\theta.y), d^0_{\theta(i)}, (\theta.x)\big)$. By definition, $\mathsf{ev}(\alpha') = P$.
- If $\alpha = \big(x,\, d_i^1,\, x\big)$, we proceed similarly by taking $\alpha' = \big((\theta.x), d^0_{\theta(i)}, (d_i\theta.y)\big)$
- If $\alpha = \alpha_1 * \alpha_2$ is a concatenation where α_1 and α_2 are shorter than α and $P \cong \mathsf{ev}(\alpha)$ via an isomorphism f, write P_i for $i = 1, 2$ for the ipomsets such that $P_i = f(\alpha_i)$. By the induction hypothesis, there exist α_1' and α_2' lifting of α_1 and α_2 such that $\mathsf{ev}(\alpha_i') = P_i$. Take $\alpha' = \alpha_1' * \alpha_2'$, so that $\mathsf{ev}(\alpha') = \mathsf{ev}(\alpha_1' * \alpha_2') = P_1 * P_2 = P$.

◀

▶ **Lemma 39.** *Let X and Y be precubical sets, and let $\alpha \in \mathrm{Path}_X$ and $\beta \in \mathrm{Path}_Y$. If $\mathsf{ev}(\alpha) \cong \mathsf{ev}(\beta)$, then there exist $\alpha' \in \mathrm{Path}_{SX}$ lifting of α and $\beta' \in \mathrm{Path}_{SY}$ lifting of β such that $\mathsf{ev}(\alpha') = \mathsf{ev}(\beta')$.*

Proof. Apply Lemma 38 for $P = \mathsf{ev}(\beta)$. ◀

Now we align paths up to congruence while preserving matching events.

▶ **Lemma 40.** *Let X and Y be (s)precubical sets, and let $\alpha \in \mathrm{Path}_X$ and $\beta \in \mathrm{Path}_Y$. If $\mathsf{ev}(\alpha) = \mathsf{ev}(\beta)$ then there exists a path $\gamma \simeq \alpha \in \mathrm{Path}_X$ such that $\gamma \equiv \beta$.*

Proof. We employ induction on the length of α.

- If $m = 0$, take $\gamma := \alpha$; congruence and matching are trivial. If $m = 1$, the condition $\mathsf{ev}(\alpha) = \mathsf{ev}(\beta)$ forces the face index and polarity to coincide, so $\alpha \equiv \beta$ and we take $\gamma := \alpha$.
- If $\alpha = (x_1, d_i^k, x_2, d_j^k, x_3)$ for $k = 0, 1$, then since $\mathsf{ev}(\alpha) = \mathsf{ev}(\beta)$, $\beta = (y_1, d_r^k, y_2, d_l^k, y_3)$ such that $\mathsf{ev}(x_s) = \mathsf{ev}(y_s)$ for $s = 1, 3$. Thus, there are two cases:
  - if $i = r$ and $j = l$ then take $\gamma = \alpha$.
  - If $i = l$ and $j = r$ then take the replacement segment (exists and unique as detailed in Definition 28) $\gamma = (y_1, d_l^k, y_2', d_r^k, y_3)$.
- If $\alpha = \alpha_1 * \alpha_2 * \alpha_3$ and $\beta = \alpha_1 * \beta_2 * \alpha_3$ such that $\alpha_2 \simeq \beta_2$. The required condition follows immediately from the induction hypothesis. ◀

Combining the previous constructions, ipomset equivalence can be lifted to stepwise agreement.

▶ **Lemma 41.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be HDAs, and let $\alpha \in \mathrm{Path}_{\mathcal{X}}$ and $\beta \in \mathrm{Path}_{\mathcal{Y}}$. If $\mathsf{ev}(\alpha) \cong \mathsf{ev}(\beta)$, then there exist $\gamma \in \mathrm{Path}_{S\mathcal{X}}$ lifting of α, $\beta' \in \mathrm{Path}_{S\mathcal{Y}}$ lifting of β, and $\alpha' \cong \gamma$ such that ST-trace(α') $=ST$-trace(β').*

Proof. Let $\beta' = S\beta$. By Lemma 39, there exists $\gamma \in \mathrm{Path}_{S\mathcal{X}}$ lifting of α such that $\mathsf{ev}(\gamma) = \mathsf{ev}(\beta)$. By Lemma 40, there exists $\alpha' \simeq \gamma$ such that $\alpha' \equiv \beta'$. By Lemma 37, ST–trace(α') =ST–trace(β'). ◀

We can now return to the original HDAs.

▶ **Proposition 42.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be HDAs, and let $\alpha \in \mathrm{Path}_{\mathcal{X}}$ and $\beta \in \mathrm{Path}_{\mathcal{Y}}$. If $\mathsf{ev}(\alpha) \cong \mathsf{ev}(\beta)$, then there exists $\gamma \simeq \alpha$ such that ST-trace(γ) =ST-trace(β).*

Proof. By Lemma 41, there exist $\alpha' \in \mathrm{Path}_{S\mathcal{X}}$ lifting of α, $\beta' \in \mathrm{Path}_{S\mathcal{X}}$ lifting of β, and $\gamma' \cong \alpha'$ such that ST-trace(γ') =ST-trace(β'). Further, by [17, Prop. 5.2], there exists $\gamma \in \mathrm{Path}_X$ such that $\gamma \simeq \alpha$ and γ' lifting of γ. On the one hand we have, ST-trace(γ') =ST-trace(β'). On the other hand, by Proposition 33, ST–trace(γ') =ST–trace(γ) and ST–trace(β) =ST–trace(β'). Thus, ST–trace(β) =ST–trace(γ). ◀

8 Bisimulations for Higher Dimensional Automata

We reformulate history preserving and hereditary history preserving bisimulation for HDAs inipomset terms, connecting them to open maps [16] and language semantics [10].

▶ **Definition 43.** *A* history preserving bisimulation *(hp-bisimulation) between HDAs $\mathcal{Y}$ and $\mathcal{Z}$ is a symmetric relation R between paths in Y and Z such that*

1. *initial paths (i_Y) and (i_Z) are related;*
2. *for all $(\rho, \sigma) \in R$, ST–trace(ρ) = ST-trace(σ);*
3. *for all $(\rho, \sigma) \in R$ and path ρ' in Y such that ρ and ρ' may be concatenated, there exists a path σ' in Z such that $(\rho * \rho', \sigma * \sigma') \in R$;*
4. *for all $(\rho, \sigma) \in R$ and path ρ' in Y such that $\rho \overset{\ell}{\rightsquigarrow} \rho'$, there exists a path σ' in Z such that $\sigma \overset{\ell}{\rightsquigarrow} \sigma'$ and $(\rho', \sigma') \in R$;*

The relation R is called hereditary history preserving bisimulation *(hhp-bisimulation) if, in addition, it satisfies:*

5. *for all $(\rho, \sigma) \in R$ and ρ' restriction of ρ, there exists σ' restriction of σ such that $(\rho', \sigma') \in R$.*

We say that $\mathcal{X}$ and $\mathcal{Y}$ are (hereditary) history preserving bisimilar *and write $\mathcal{X} \approx_{(h)hp} \mathcal{Y}$ if there exists a (hereditary)* history- preserving bisimulation *R between them; this is an equivalence relation.*

ST-bisimulation is defined as the hp-bisimulation but dropping clauses 4 and 5.

▶ **Theorem 44.** *Two HDAs $\mathcal{X}$ and $\mathcal{Y}$ are ST-bisimilar* iff *there exists $R \subseteq \mathrm{Path}_{\mathcal{X}} \times \mathrm{Path}_{\mathcal{Y}}$ such that:*

1. *initial paths (i_Y) and (i_Z) are related;*
2. $(\alpha, \beta) \in R \implies \mathsf{ev}(\alpha) \cong \mathsf{ev}(\beta)$;
3. *R respects path initial inclusion: for all $(\rho, \sigma) \in R$ and path ρ' in Y such that ρ and ρ' may be concatenated, there exists a path σ' in Z such that $(\rho * \rho', \sigma * \sigma') \in R$;*

Proof. "⇒" Let R be an ST–bisimulation between $\mathcal{X}$ and $\mathcal{Y}$. By Lemma 35, R satisfies the three conditions above.

"⇐" Let K be a relation satisfying the stated conditions. By Proposition 42, K induces a relation $R = \{ (\alpha', \beta) \mid (\alpha, \beta) \in K,\ \alpha' \simeq \alpha,\ \text{ST–trace}(\alpha') = \text{ST–trace}(\beta) \}$.

By construction, R is an ST–bisimulation relating $\mathcal{X}$ and $\mathcal{Y}$. ◀

▶ **Theorem 45.** *Two HDAs $\mathcal{X}$ and $\mathcal{Y}$ are* (h)hp*–bisimilar* iff *there exists a relation $R \subseteq \mathrm{Path}_{\mathcal{X}} \times \mathrm{Path}_{\mathcal{Y}}$ that satisfies conditions* 1, 3, 4, (5) *of the hhp–bisimulation definition,* and *the following replacement of clause 2: $(\alpha, \beta) \in R \implies \mathsf{ev}(\alpha) \cong \mathsf{ev}(\beta)$.*

Proof. ($\Rightarrow$) Let R be an (h)hp–bisimulation between $\mathcal{X}$ and $\mathcal{Y}$. Arguing as in the proof of Th 44, R satisfies the stated conditions, with ipomset isomorphism replacing equality of ST–traces.

($\Leftarrow$) Let K be a relation satisfying the stated conditions. Construct the relation R as in the proof of Th 44. For clause 4 of hp–bisimulation, Since $(\rho, \sigma) \in R$, there exists α such that $\rho \simeq \alpha$, $(\alpha, \sigma) \in K$, and ST-trace(ρ) = ST-trace(σ). By Definition 29, the congruence $\rho \simeq \alpha$ is generated exclusively by the reversible rules (1) and (2) of Definition 28.

Step 1: σ is ℓ-swappable. Since ST-trace(ρ) = ST-trace(σ), the polarity of the face maps at positions ℓ and $\ell + 1$ coincides in ρ and σ. *Rules (1) and (2).* The ST-trace equality forces σ to have two consecutive face maps of the same polarity at positions ℓ and $\ell + 1$. Since rules (1) and (2) are reversible and apply to any two consecutive same-polarity steps regardless of index ordering, σ is ℓ-swappable, yielding $\sigma \overset{\ell}{\rightsquigarrow} \sigma'$. *Rules (3) and (4).* The ST-trace records start$(i_{\ell+1})$ explicitly at every down-step. Since ST-trace(ρ) = ST-trace(σ), we have $\mathrm{start}_\rho(i_{\ell+1}) = \mathrm{start}_\sigma(r_{\ell+1})$. Hence the directed swap condition holds for σ at position ℓ, giving $\sigma \overset{\ell}{\leftrightsquigarrow} \sigma'$.

Step 2: $(\rho', \sigma') \in R$. We must find α' such that $\rho' \simeq \alpha'$, $(\alpha', \sigma') \in K$, and ST-trace(ρ') = ST-trace(σ'). We proceed by induction on the length n of the congruence chain from ρ to α. *Base case $n = 0$, i.e. $\rho = \alpha$.* Then $\alpha \overset{\ell}{\leftrightsquigarrow} \rho'$ is the same swap as on ρ. Clause (4) of K applied to $(\alpha, \sigma) \in K$ yields $\sigma \overset{\ell}{\leftrightsquigarrow} \sigma'$ and $(\alpha', \sigma') \in K$ where $\alpha' = \rho'$. Hence $\rho' \simeq \alpha'$ trivially and ST-trace(ρ') = ST-trace(σ') by the ST-trace analysis of Step 1. Thus $(\rho', \sigma') \in R$. *Inductive step $n > 0$.* By Definition 29, we can write $\rho = \gamma_1 * \rho_0 * \delta$ and $\alpha = \gamma_1 * \alpha_0 * \delta$, where $\rho_0 \simeq \alpha_0$ via a congruence chain of length $n - 1$, using the third clause of Definition 29. We distinguish three sub-cases according to the position of the ℓ-swap.

- *Swap inside γ_1 or inside δ.* The sub-segment ρ_0 is unaffected. The result ρ' has the same decomposition with ρ_0 unchanged, so $\rho' \simeq \alpha'$ where α' is obtained by applying the same swap to α, and $(\alpha', \sigma') \in K$ by clause (4) of K.
- *Swap entirely inside ρ_0.* The induction hypothesis applies directly to ρ_0 and α_0, yielding α'_0 with $\rho'_0 \simeq \alpha'_0$ and $(\gamma_1 * \alpha'_0 * \delta, \sigma') \in K$. Setting $\alpha' := \gamma_1 * \alpha'_0 * \delta$ gives $\rho' \simeq \alpha'$.
- *Swap at the boundary between ρ_0 and δ (or between γ_1 and ρ_0).* Absorb the two boundary steps into an extended segment $\rho_0^+ = \rho_0 *$ (first step of δ) and $\delta^- = \delta$ minus its first step, so that $\rho = \gamma_1 * \rho_0^+ * \delta^-$ and the swap at ℓ now falls entirely inside ρ_0^+. The induction hypothesis applies to ρ_0^+ and α_0^+ (defined analogously), reducing to the previous sub-case.

In all cases we obtain α' with $\rho' \simeq \alpha'$, $(\alpha', \sigma') \in K$, and ST-trace(ρ') = ST-trace(σ'), so $(\rho', \sigma') \in R$. In the hereditary case, clause 5 follows directly by construction. Hence R is an (h)hp–bisimulation. ◀

9 Conclusion

We have developed an order-free semantic foundation for higher-dimensional automata. The central technical contributions are the categorical isomorphism between HDAs over the unordered base Ξ and symmetric HDAs, the canonical assignment of interval ipomsets to execution paths, the formal correspondence between ipomset labels and ST-traces, and the characterization of ST- and hhp-bisimulation via ipomset isomorphism. These results have concrete consequences beyond what is explicitly developed here.

On the logical side, the order-free path-category structure established here provides the missing foundation for deriving, via the Open Maps framework [16], modal logics that canonically characterize hhp-bisimulation; the temporal and modal logics of [5, 2, 25] are natural targets for revisiting under this symmetric foundation. On the structural side, the elimination of the event order artifact opens a cleaner path toward systematic translations between HDAs and other models of concurrency such as Petri nets, where the mismatches documented in [4, 6] were a direct consequence of the representational incompatibility resolved here.

References

1 Amazigh Amrane, Hugo Bazille, Emily Clement, and Uli Fahrenberg. Languages of higher-dimensional timed automata. In Lars Michael Kristensen and Jan Martijn van der Werf, editors, *Application and Theory of Petri Nets and Concurrency*, pages 197–219, Cham, 2024. Springer Nature Switzerland. `doi:10.1007/978-3-031-61433-0_10`.

2 Amazigh Amrane, Hugo Bazille, Emily Clement, Uli Fahrenberg, Marie Fortin, and Krzysztof Ziemiański. Büchi-elgot-trakhtenbrot theorem for higher-dimensional automata, 2025. `doi:10.48550/arXiv.2505.10461`.

3 Amazigh Amrane, Hugo Bazille, Uli Fahrenberg, and Marie Fortin. Logic and languages of higher-dimensional automata. In Joel D. Day and Florin Manea, editors, *Developments in Language Theory*, pages 51–67, Cham, 2024. Springer Nature Switzerland. `doi:10.1007/978-3-031-66159-4_5`.

4 Amazigh Amrane, Hugo Bazille, Uli Fahrenberg, Loïc Hélouët, and Philipp Schlehuber-Caissier. Petri nets and higher-dimensional automata. In Elvio Amparore and Łukasz Mikulski, editors, *Application and Theory of Petri Nets and Concurrency*, pages 18–40, Cham, 2025. Springer Nature Switzerland. `doi:10.1007/978-3-031-94634-9_2`.

5 Amazigh Amrane, Hugo Bazille, Uli Fahrenberg, and Krzysztof Ziemiański. Closure and decision properties for higher-dimensional automata. *Theoretical Computer Science*, 1036:115156, 2025. `doi:10.1016/j.tcs.2025.115156`.

6 Amazigh Amrane, Hugo Bazille, Timothée Fragnaud, and Philipp Schlehuber-Caissier. Towards an efficient conversion of petri nets into higher dimensional automata. In *Modélisation des Systèmes Réactifs (MSR'25)*, pages 1–15, Reims, France, November 2025. Modélisation des Systèmes Réactifs. URL: `https://hal.science/hal-05281114`.

7 Marek A. Bednarczyk. *Categories of Asynchronous Systems*. PhD thesis, University of Sussex, GBR, 1987. AAIDX83002.

8 Emily Clement, Enzo Erlich, and Jérémy Ledent. Kamp theorem for pomset languages of higher dimensional automata, 2025. `arXiv:2410.12493`.

9 Uli Fahrenberg, Christian Johansen, Georg Struth, and Ratan Bahadur Thapa. Generating posets beyond n. In Uli Fahrenberg, Peter Jipsen, and Michael Winter, editors, *Relational and Algebraic Methods in Computer Science*, pages 82–99, Cham, 2020. Springer International Publishing. `doi:10.1007/978-3-030-43520-2_6`.

10 Uli Fahrenberg, Christian Johansen, Georg Struth, and Krzysztof Ziemiański. Languages of higher-dimensional automata. *Math. Struct. Comput. Sci.*, 31(5):575–613, 2021. `doi:10.1017/S0960129521000293`.

11 Uli Fahrenberg, Christian Johansen, Georg Struth, and Krzysztof Ziemianski. Kleene theorem for higher-dimensional automata. *Log. Methods Comput. Sci.*, 20(4):22:1–22:50, 2024. `doi:10.46298/LMCS-20(4:22)2024`.

12 Peter C Fishburn. Interval orders and interval graphs: A study of partially ordered sets. *(No Title)*, 1985.

13 Jay L. Gischer. The equational theory of pomsets. *Theoretical Computer Science*, 61(2):199–224, 1988. doi:10.1016/0304-3975(88)90124-7.

14 Jan Grabowski. On partial languages. *Fundamenta Informaticae*, 4(2):427–498, 1981.

15 Ryszard Janicki and Maciej Koutny. Structure of concurrency. *Theoretical Computer Science*, 112(1):5–52, 1993. doi:10.1016/0304-3975(93)90238-O.

16 André Joyal, Mogens Nielsen, and Glynn Winskel. Bisimulation from open maps. *Information and Computation*, 127(2):164–185, 1996. doi:10.1006/inco.1996.0057.

17 Thomas Kahl. On symmetric higher-dimensional automata and bisimilarity. *Theoretical Computer Science*, 935:47–60, 2022. doi:10.1016/j.tcs.2022.07.004.

18 Mogens Nielsen, Gordon Plotkin, and Glynn Winskel. Petri nets, event structures and domains, part i. *Theoretical Computer Science*, 13(1):85–108, 1981. doi:10.1016/0304-3975(81)90112-2.

19 Vaughan Pratt. Modeling concurrency with geometry. In *Proceedings of the 18th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL '91, pages 311–322, New York, NY, USA, 1991. Association for Computing Machinery. doi:10.1145/99583.99625.

20 Vaughan R. Pratt. Modeling concurrency with partial orders. *International Journal of Parallel Programming*, 15:33–71, 1986. doi:10.1007/BF01379149.

21 Mike W Shields. Concurrent machines. *The Computer Journal*, 28(5):449–465, 1985. doi:10.1093/COMJNL/28.5.449.

22 Georg Struth and Krzysztof Ziemiański. Presheaf automata, 2024. arXiv:2409.04612.

23 Rob J. van Glabbeek. On the expressiveness of higher dimensional automata. *Theoretical Computer Science*, 356(3):265–290, 2006. Expressiveness in Concurrency. doi:10.1016/j.tcs.2006.02.012.

24 Rob J. van Glabbeek and Gordon D. Plotkin. Configuration structures. In *Proceedings of Tenth Annual IEEE Symposium on Logic in Computer Science*, pages 199–209, San Deigo, CA, USA, June 1995. IEEE. doi:10.1109/LICS.1995.523257.

25 Safa Zouari, Krzysztof Ziemiański, and Uli Fahrenberg. Bisimulations and logics for higher-dimensional automata. In Chutiporn Anutariya and Marcello M. Bonsangue, editors, *Theoretical Aspects of Computing – ICTAC 2024*, pages 132–150, Cham, 2025. Springer Nature Switzerland.