Sesqui-Pushout Rewriting:
Concurrency, Associativity and Rule Algebra Framework[*]

Nicolas Behr

Université de Paris, IRIF, CNRS
F-75013 Paris, France
`nicolas.behr@irif.fr`

Sesqui-pushout (SqPO) rewriting is a variant of transformations of graph-like and other types of structures that fit into the framework of adhesive categories where deletion in unknown context may be implemented. We provide the first account of a concurrency theorem for this important type of rewriting, and we demonstrate the additional mathematical property of a form of associativity for these theories. Associativity may then be exploited to construct so-called rule algebras (of SqPO type), based upon which in particular a universal framework of continuous-time Markov chains for stochastic SqPO rewriting systems may be realized.

1 Motivation and relation to previous work

The framework of *Sesqui-Pushout (SqPO) rewriting* has been introduced relatively recently in [16] as a novel alternative to the pre-existing algebraic graph transformation frameworks known as *Double-Pushout (DPO)* [26, 24, 17, 34] and *Single-Pushout (SPO) rewriting* [32, 35, 38]. In the setting of the rewriting of graph-like structures, the distinguishing feature of the aforementioned DPO-type rewriting is that the deletion of vertices with incident edges is only possible if the incident edges are explicitly deleted via the application of the rewriting rule. In contrast, in both the SqPO and the SPO rewriting setups, *"deletion in unknown context"* is implementable. Thus for practical applications of rewriting, in particular in view of the modeling of stochastic rewriting systems, the S(q)PO rewriting semantics provide an important additional option for the practitioners, and will thus in particular complement the existing DPO-type associative rewriting and rule algebra framework as introduced in [3, 7]. Referring the interested readers to [36] for a recent review and further conceptual details of the three approaches, suffice it here to quote that SqPO and SPO rewriting via linear rules[1] (defined as monic spans) and along monomorphic matches effectively encode the same semantics of rewriting. We chose (by the preceding argument without loss of expressivity) to develop the theory of associative rewriting within the SqPO rather than the SPO setting, since the SqPO framework bears certain close technical similarities to the DPO rewriting framework, which proved crucial in finding a strategy for the highly intricate proofs of the concurrency and associativity theorems presented in this paper. While it is well-known (see e.g. Section 5.1 of [16]) that DPO- and SqPO-type semantics coincide for certain special classes of linear rules (essentially rules that do not delete vertices), and while these cases might provide some valuable cross-checks of technical results to the experts, SqPO-type semantics is in its full generality a considerably more intricate variant of semantics due to its inherent "mixing" of pushouts with final pullback complements. It should further be noted that we must impose a set of additional assumptions on the underlying

---

[1]While non-linear rules in SqPO rewriting have interesting applications in their own right (permitting e.g. the cloning and fusing of vertices in graphs), this most general case is left for future work.

adhesive categories (see Assumption 1) in order to ensure certain technical properties necessary for our concurrency and associativity theorems to hold. To the best of our knowledge, apart from some partial results in the direction of developing a concurrency theorem for SqPO-type rewriting in [16, 36, 15], prior to this work neither of the aforementioned theorems had been available in the SqPO framework.

Associativity of SqPO rewriting theories plays a pivotal role in our development of a novel form of concurrent semantics for these theories, the so-called *SqPO-type rule algebras*. Previous work on associative DPO-type rewriting theories [3, 5, 7] (see also [8]) has led to a category-theoretical under-standing of associativity that may be suitably extended to the SqPO setting. In contrast to the traditional and well-established formalisms of concurrency theory for rewriting systems (see e.g. [42, 25, 23, 15] for DPO-type semantics and [16, 15] for a notion of parallel independence and a Local Church-Rosser theorem for SqPO-rewriting of graphs), wherein the focus of the analysis is mostly on *derivation traces* and their sequential independence and parallelism properties, the focus of our rule-algebraic approach differs significantly: we propose instead to put *sequential compositions of linear rules* at the center of the analysis (rather than the derivation traces), and moreover to employ a vector-space based semantics in order to encode the non-determinism of such rule compositions. It is for this reason that the concurrency theorem plays a quintessential role in our rule algebra framework, in that it encodes the relationship between sequential compositions of linear rules and derivation traces, which in turn gives rise to the so-called *canonical representations* of the rule algebras (see Section 4). This approach in particular permits to uncover certain combinatorial properties of rewriting systems that would otherwise not be accessible. While undoubtedly not a standard technique in the realm of theoretical computer science, certain special examples of rule algebras are ubiquitous in many areas of applied mathematics and theoretical physics. The most famous such example concerns the so-called *Heisenberg-Weyl algebra* (see e.g. [9, 10, 11]), which is well-known to possess a representation in terms of the formal multiplication operator $\hat{x}$ and the differentiation operator ∂_x on formal power series in the formal variable x, with $\hat{x}x^n := x^{n+1}$ and ∂_x acting as the derivative. Referring the interested readers to Example 2 (see also [7, 4]) for the precise details, it transpires that the monomials x^n (for n a non-negative integer) are found to be in one-to-one correspondence with *graph states* associated to n-vertex discrete graphs, while $\hat{x}$ and ∂_x may be understood as the *canonical representations* of the discrete graph rewriting rules of creation and deletion of vertices. It will thus come as no surprise that considering more general rewriting rules than those of discrete graphs will lead to a very substantial generalization of these traditional results and techniques.

From the very beginning of the development of the rule algebra framework [3], one of our main motivations has been the study of stochastic rewriting systems, whence of continuous-time Markov chains (CTMCs) based upon DPO- or SqPO-type rewriting rules. While previously in particular applications of stochastic SqPO-type rewriting systems have played a role predominantly in highly specialized settings such as e.g. the formulation of the biochemical reaction system framework known as KAPPA [20, 21, 19, 18], our novel approach of formulating such systems in terms of associative unital rule algebras may very well open this versatile modeling technique to many other areas of applied research. In conjunction with our previously developed DPO-type framework in [7], one could argue that our *stochastic mechanics frameworks* are in a certain sense a *universal construction*, in that once a semantics for associative unital rewriting is provided, the steps necessary to obtain the associated CTMCs are clearly formalized. It is interesting to compare the traditional approaches to stochastic rewriting systems with CTMC semantics such as [31, 30] in the DPO- and [29] in the SPO-settings to our present reformulation in terms of rule algebras. The former approaches yet again tend to focus on derivation traces of stochastic rewriting systems, while our rule-algebraic approach aims to extract dynamical in-

formation from stochastic rewriting systems via analysis of certain combinatorial relationships (so-called nested commutators) of the infinitesimal generator of the CTMC with the (pattern-counting) observables of the system. It is via these relations that one may in certain cases obtain *exact closed-form solutions* for such dynamical data (see e.g. Section 6). It would nevertheless be an intriguing avenue for future research to understand better the finer points of the "traditional" stochastic rewriting frameworks (which also feature sophisticated developments in terms of probabilistic model-checking and various types of stochastic logics), and furthermore whether or not rule-algebraic techniques might be of interest also in more general stochastic rewriting semantics such as probabilistic (timed) graph transformations [33, 40].

Structure of the paper: In Section 2, some category-theoretical background material is provided. The key results of associativity and concurrency of SqPO rewriting are presented in Section 3, followed by the construction of SqPO-type rule algebras in Section 4. The second part of the paper contains the stochastic mechanics framework (Section 5) as well as a practical application example (Section 6). Technical proofs are situated in the Appendix.

2   Background: adhesive categories and final pullback complements

We recall some of the elementary definitions and properties related to the notions of adhesive categories, upon which our framework will rely.

Definition 1 ([34])**.** A category $\mathbf{C}$ is said to be ***adhesive*** if

   (i) $\mathbf{C}$ has pushouts along monomorphisms,

  (ii) $\mathbf{C}$ has pullbacks,

 (iii) pushouts along monomorphisms are van Kampen (VK) squares.

The last property entails that in a commutative cube as in (1) on the left where the bottom square is a pushout, this square is a VK square if and only if whenever the back and right vertical faces are pullbacks, then the top square is a pushout if and only if the front and left vertical squares are pullbacks.

$$(1)$$

We will be exclusively interested in categories that satisfy certain finiteness properties (in order to ensure finiteness of the sets of matches for rule applications and compositions, see Section 3):

Definition 2 (Finitary categories [12])**.** A category $\mathbf{C}$ is said to be *finitary* if every object $X \in \mathrm{obj}(\mathbf{C})$ has only finitely many subobjects (i.e. if there only exist finitely many monomorphisms $Y \to X$ up to isomorphism for every $X \in \mathrm{obj}(\mathbf{C})$). For every adhesive category $\mathbf{C}$, the restriction to finite objects of $\mathbf{C}$ defines a full subcategory $\mathbf{C}_{fin}$ called the *finitary restriction* of $\mathbf{C}$.

Theorem 1 (Finitary restrictions; [12], Thm. 4.6)**.** *The finitary restriction $\mathbf{C}_{fin}$ of any adhesive category $\mathbf{C}$ is a finitary adhesive category.*

Adhesive categories have been introduced and advocated in [34] as a framework for rewriting due to their numerous useful properties, some of which are listed in Appendix A for the reader's convenience. One of the central concepts in the theory of SqPO rewriting is the following:

Definition 3 (Final Pullback Complement (FPC); [16, 36])**.** Let $\mathbf{C}$ be a category. Given a commutative diagram as in (1) on the right, a pair of morphisms (d, b) is a *final pullback complement (FPC)* of a pair (c, a) if (i) (a, b) is a pullback of (c, d) (i.e. if the square marked (B) is a pullback square), and (ii) for each collection of morphisms (x, y, z, w) as in (1) on the right, where (x, y) is pullback of (c, z) and where $a \circ w = x$, there exists a unique morphism w^* with $d \circ w^* = z$ and $w^* \circ y = b \circ w$.

For our associative rewriting framework, it will be crucial to work with a category in which (i) FPCs are guaranteed to exist when constructing them for composable pairs of monomorphisms, and (ii) monomorphisms are stable under FPCs, i.e. FPCs of pairs of monomorphisms are given by pairs of monomorphisms. This property is satisfied by adhesive categories (cf. Lemma 3 of Appendix A), yet to the best of our knowledge the question of which more general types of categories possess this property has not yet been investigated to quite the level of generality as analogous classification problems in the case of DPO rewriting, even though there does exist a large body of work on classes of categories that admit SqPO constructions [16, 39, 36, 14, 37]. Within these classes, according to [13, 14] guarantees for the existence of FPCs may be provided for categories that possess a so-called $\mathscr{M}$-partial map classifier. However, it appears to be an open question of whether the statement of Lemma 3 on stability of monomorphisms under FPCs may be generalized to the setting of $\mathscr{M}$-adhesive categories, where $\mathscr{M}$ is a class of monomorphisms. Relaying such questions to future work, we refer to Lemma 4 of Appendix A for a well-known instantiation of a suitable categorical setting from the SqPO literature in the form of the category **FinGraph** of finite directed multigraphs, which also serves to illustrate the FPC construction.

Assumption 1. $\mathbf{C}$ *is an adhesive category in which all FPCs along monomorphisms exist, and in which monomorphisms are stable under FPCs.*

3   Sesqui-Pushout rewriting

We will now develop a framework for *Sesqui-Pushout (SqPO) rewriting* in the setting of a category $\mathbf{C}$ satisfying Assumption 1, in close analogy to the framework of associative Double-Pushout (DPO) rewriting as introduced in [7, 8]. Unlike in the general setting of SqPO rewriting, we will thus be able to not only prove a *concurrency theorem* (Section 3.1), but also an *associativity property* of the SqPO-type rule composition (Section 3.2).

3.1   Concurrent composition and concurrency theorem

For reasons that will become more transparent when introducing the SqPO-type rule algebra framework starting from Section 4, we opt for a non-standard convention of reading spans of monomorphisms "from right to left" (rather than the traditional "left to right"), which is why we will speak of "input" and "output" of rules rather than "left-" and "right hand sides" to avoid confusion.

Definition 4 (SqPO-type rewriting; compare [16], Def. 4)**.** Let $\mathbf{C}$ be an adhesive category satisfying Assumption 1. Denote by $\mathrm{Lin}(\mathbf{C})$ the set of (isomorphism classes[2] of) so-called *linear productions*,

---

[2]Two productions $O \leftarrow K \rightarrow I$ and $O' \leftarrow K' \rightarrow I'$ are defined to be isomorphic if there exist isomorphisms $I \rightarrow I'$, $K \rightarrow K'$ and $O \rightarrow O'$ that make the obvious diagram commute; we will not distinguish between isomorphic productions. As natural in this category-theoretical setting, the constructions presented in the following are understood as defined up to such isomorphisms.

defined as the set of spans of monomorphisms,

$$\mathsf{Lin}(\mathbf{C}) := \{p \equiv (O \xleftarrow{o} K \xrightarrow{i} I) \mid o, i \in \mathsf{mono}(\mathbf{C})\} \big/ \cong. \tag{2}$$

Given an object $X \in \mathsf{obj}(\mathbf{C})$ and a linear production $p \in \mathsf{Lin}(\mathbf{C})$, we denote the *set of SqPO-admissible matches* $\mathsf{M}_p^{sq}(X)$ as the set of monomorphisms $m : I \to X$. Then the diagram below is constructed by taking the *final pullback complement* marked FPC followed by taking the pushout marked PO:

$$
\begin{array}{ccccc}
O & \xleftarrow{\;o\;} & K & \xrightarrow{\;i\;} & I \\
{\scriptstyle m^*}\downarrow & \text{PO} \; {\scriptstyle k}\downarrow & & \text{FPC} \;\; \downarrow{\scriptstyle m} \\
X' & \xleftarrow{\;o'\;} & \overline{K} & \xrightarrow{\;i'\;} & X
\end{array}
\tag{3}
$$

We write $p_m(X) := X'$ for the object "produced" by the above diagram. The process is called *(SqPO-) derivation* of X along production p and admissible match m, and denoted $p_m(X) \overset{SqPO}{\underset{p,m}{\Longleftarrow}} X$.

Next, a notion of sequential composition of productions is introduced:

Definition 5 (SqPO-type concurrent composition). Let $p_1, p_2 \in \mathsf{Lin}(\mathbf{C})$ be two linear productions. Then an overlap of the output object O_1 of p_1 with the input object I_2 of p_2, encoded as a span

$$\mathbf{m} = (I_2 \xleftarrow{m_2} M_{21} \xrightarrow{m_1} O_1)$$

with $m_1, m_2 \in \mathsf{mono}(\mathbf{C})$, is called an *SqPO-admissible match of p_2 into p_1*, denoted $\mathbf{m} \in \mathsf{M}_{p_2}^{sq}(p_1)$, if the square marked POC in (4) is constructible as a pushout complement (with the cospan $I_2 \xrightarrow{n_2} N_{21} \xleftarrow{n_1} O_1$ obtained by taking the pushout marked PO). In this case, the remaining parts of the diagram are formed by taking the final pullback complement marked FPC and the pushouts marked PO:

$$\tag{4}$$

If $\mathbf{m} \in \mathsf{M}_{p_2}^{sq}(p_1)$, we write $p_2 \overset{\mathbf{m}}{\lhd} p_1 \in \mathsf{Lin}(\mathbf{C})$ for the *composite* of p_2 with p_1 along the admissible match $\mathbf{m}$, defined as

$$p_2 \overset{\mathbf{m}}{\lhd} p_1 \equiv (O_{21} \xleftarrow{o_{21}} K_{21} \xrightarrow{i_{21}} I_{21}). \tag{5}$$

Due to stability of monomorphisms under pushouts, pullbacks and FPCs in the setting of a category satisfying Assumption 1, all morphisms in Definitions 4 and 5 are guaranteed to be monomorphisms, whence in particular the span $p_2 \overset{\mathbf{m}}{\lhd} p_1$ is a span of monomorphisms and thus indeed an element of $\mathsf{Lin}(\mathbf{C})$.

At first sight, it might appear irritating that in the definition of the SqPO-type rule composition, the right hand part of (4) involves a *pushout complement* (marked POC), while the left hand part of the diagram in (4) features a *final pullback complement* (marked FPC). Intuitively, considering the case of graph rewriting for concreteness, in a given sequential application of two productions, while the application of the first production may lead to implicit edge deletions, the second production is incapable

of having any causal interaction with edges deleted by the first production. In contrast, the second production may in a given sequential application very well implicitly delete edges present in the output object of the first production, which explains the presence of the FPC in the defining equation (4). We refer the interested readers to [5] for further intuitions attainable in terms of so-called rule diagrams for presenting rule compositions. The justification for Definition 5 in the general case is provided via the following *concurrency theorem*. Even though at least in certain specialized settings the "synthesis" part of this theorem has been foreseen already in [36] (where it is also commented that a full concurrency theorem for SqPO rewriting might be attainable), the following result appears to be new.

Theorem 2 (SqPO-type Concurrency Theorem). *Let* $\mathbf{C}$ *be an adhesive category satisfying Assumption 1. Let* $p_1, p_2 \in \mathsf{Lin}(\mathbf{C})$ *be two linear rules and* $X_0 \in ob(\mathbf{C})$ *an object.*

- **Synthesis:** *Given a two-step sequence of SqPO derivations*

$$X_2 \xLeftarrow[p_2,m_2]{SqPO} X_1 \xLeftarrow[p_1,m_1]{SqPO} X_0 \,,$$

   with $X_1 := p_{1_{m_1}}(X_0)$ *and* $X_2 := p_{2_{m_2}}(X_1)$, *there exists a SqPO-composite rule* $q = p_2 \overset{\mathbf{n}}{\lhd} p_1$ *for a unique* $\mathbf{n} \in \mathbf{M}^{sq}_{p_2}(p_1)$, *and a unique SqPO-admissible match* $n \in \mathsf{M}^{sq}_q(X)$, *such that*

$$q_n(X) \xLeftarrow[q,n]{SqPO} X_0 \qquad and \qquad q_n(X_0) \cong X_2 \,.$$

- **Analysis:** *Given an SqPO-admissible match* $\mathbf{n} \in \mathbf{M}^{sq}_{p_2}(p_1)$ *of* p_2 *into* p_1 *and an SqPO-admissible match* $n \in \mathsf{M}^{sq}_q(X)$ *of the SqPO-composite* $q = p_2 \overset{\mathbf{n}}{\lhd} p_1$ *into* X, *there exists a unique pair of SqPO-admissible matches* $m_1 \in \mathsf{M}^{sq}_{p_1}(X_0)$ *and* $m_2 \in \mathsf{M}^{sq}_{p_2}(X_1)$ *with* $X_1 := p_{1_{m_1}}(X_0)$ *such that*

$$X_2 \xLeftarrow[p_2,m_2]{SqPO} X_1 \xLeftarrow[p_1,m_1]{SqPO} X_0 \qquad and \qquad X_2 \cong q_n(X) \,.$$

Proof. See Appendix B.1.   □

3.2 Composition and associativity

The following theorem establishes that in analogy to the DPO rewriting setting of [7], also the sesqui-pushout variant of rule compositions possesses a form of associativity property.

Theorem 3 (SqPO-type associativity theorem). *Let* $\mathbf{C}$ *be an adhesive category satisfying Assumption 1. Then the SqPO-composition operation* $. \overset{.}{\lhd} .$ *on linear productions of* $\mathbf{C}$ *is* associative *in the following sense: given linear productions* $p_1, p_2, p_3 \in \mathsf{Lin}(\mathbf{C})$, *there exists a bijective correspondence between pairs of SqPO-admissible matches* $(\mathbf{m}_{21}, \mathbf{m}_{3(21)})$ *and* $(\mathbf{m}_{32}, \mathbf{m}_{(32)1})$ *such that*

$$p_3 \overset{\mathbf{m}_{3(21)}}{\lhd} \left(p_2 \overset{\mathbf{m}_{21}}{\lhd} p_1 \right) \cong \left(p_3 \overset{\mathbf{m}_{32}}{\lhd} p_2 \right) \overset{\mathbf{m}_{(32)1}}{\lhd} p_1 \,. \tag{6}$$

Proof. Intuitively, the associativity property in the SqPO case manifests itself in a form entirely analogous to the DPO case [7], whereby the data provided along the path highlighted in orange below permits to uniquely compute the data provided along the path highlighted in blue and vice versa (with both sets

of overlaps computing the same "triple composite" production that is encoded as the composition of the three spans in the bottom front row):

$$\tag{7}$$

See Appendix B.2 for the precise technical details of the proof. □

We invite the interested readers to compare the SqPO-type constructions presented here against those contained in the extended journal version [8] of [7] for the DPO framework, since this might lend some intuitions on the otherwise very abstract nature of the proofs to the experts.

4   From associativity to SqPO-type rule algebras

For the rule algebra constructions, we will require an additional structure:

Definition 6 (Initial objects). An object $\varnothing \in \mathsf{obj}(\mathbf{C})$ of some category $\mathbf{C}$ is said to be a *strict initial object* if for every object $X \in \mathsf{obj}(\mathbf{C})$, there exists a unique morphism $\varnothing \to X$, and if any morphism $X \to \varnothing$ must be an isomorphism.

For example, the category **Graph** and its finitary restriction **FinGraph** possess a strict initial object (the empty graph). For the experts, it appears worthwhile noting the following result:

Lemma 1 (Extensive categories; [34], Lem. 4.1). *An adhesive category* $\mathbf{C}$ *is an* extensive category[3] *if and only if it possesses a strict initial object.*

Assumption 2 (Prerequisites for SqPO-type rule algebras). *We assume that* $\mathbf{C}$ *is an adhesive category satisfying Assumption 1, and which is in addition* finitary *and possesses a strict initial object* $\varnothing \in \mathsf{obj}(\mathbf{C})$.

Definition 7 (SqPO-type rule algebras). Let $\delta : \mathsf{Lin}(\mathbf{C}) \to \mathscr{R}_{\mathbf{C}}$ be defined as an isomorphism from $\mathsf{Lin}(\mathbf{C})$ to the basis of a free $\mathbb{R}$-vector space $\mathscr{R}_{\mathbf{C}} \equiv (\mathscr{R}_{\mathbf{C}}, +, \cdot)$, such that[4]

$$\mathscr{R}_{\mathbf{C}} := span_{\mathbb{R}}(\{\delta(p) \mid p \in \mathsf{Lin}(\mathbf{C})\}). \tag{8}$$

In order to clearly distinguish between elements of $\mathsf{Lin}(\mathbf{C})$ and basis vectors of $\mathscr{R}_{\mathbf{C}}$, we introduce the notation

$$(O \overset{p}{\Leftarrow} I) := \delta\left(O \overset{o}{\leftarrow} K \overset{i}{\to} I\right). \tag{9}$$

Define the *SqPO rule algebra product* $\odot_{\mathscr{R}_{\mathbf{C}}}$ on a category $\mathbf{C}$ that satisfies Assumption 2 as the binary operation

$$\odot_{\mathscr{R}_{\mathbf{C}}} : \mathscr{R}_{\mathbf{C}} \times \mathscr{R}_{\mathbf{C}} \to \mathscr{R}_{\mathbf{C}} : (R_2, R_1) \mapsto R_2 \odot_{\mathscr{R}_{\mathbf{C}}} R_1, \tag{10}$$

where for two basis vectors $R_i = \delta(p_i)$ encoding the linear rules $p_i \in Lin(\mathbf{C})$ $(i = 1, 2)$,

$$R_2 \odot_{\mathscr{R}_{\mathbf{C}}} R_1 := \sum_{\mathbf{m} \in \mathbf{M}^{sq}_{p_2}(p_1)} \delta\left(p_2 \overset{\mathbf{m}}{\vartriangleleft} p_1\right). \tag{11}$$

---

[3]For the purposes of this paper, it suffices to consider the "if" direction as a definition of extensivity, since the relevant structure to our constructions is that of having a strict initial object (see e.g. [34] for the precise definition of extensivity).

[4]Recall that for a set A, the notation $span_{\mathbb{R}}(\{e(a) \mid a \in A\})$ entails to "take the $\mathbb{R}$-span over basis vectors $e(a)$ indexed by elements of A", i.e. elements of the resulting $\mathbb{R}$-vector space are (finite) linear combinations of the basis vectors $e(a)$ with real coefficients .

The definition is extended to arbitrary (finite) linear combinations of basis vectors by bilinearity, whence for $p_i, p_j \in \mathsf{Lin}(\mathbf{C})$ and $\alpha_i, \beta_j \in \mathbb{R}$,

$$\left(\sum_i \alpha_i \cdot \delta(p_i) \right) \odot_{\mathscr{R}_{\mathbf{C}}} \left(\sum_j \beta_j \cdot \delta(p_j) \right) := \sum_{i,j} (\alpha_i \cdot \beta_j) \cdot (\delta(p_i) \odot_{\mathscr{R}_{\mathbf{C}}} \delta(p_j)). \tag{12}$$

We call $\mathscr{R}_{\mathbf{C}}^{sq} \equiv (\mathscr{R}_{\mathbf{C}}, \odot_{\mathscr{R}_{\mathbf{C}}})$ the **SqPO-type rule algebra** over the finitary adhesive and extensive category **C**.

The rule algebra product $R_2 \odot_{\mathscr{R}_{\mathbf{C}}} R_1$ for $R_j = \delta(r_j)$ ($j = 1, 2$) thus encodes the non-determinism in the SqPO-type sequential composition of the linear rule r_2 with r_1 in terms of the "sum over all possible compositions". As the following example illustrates, since δ is defined to map from *isomorphism classes* of linear rules to basis vectors of $\mathscr{R}_{\mathbf{C}}$, and since two distinct matches may lead to isomorphic composite rules, $R_2 \odot_{\mathscr{R}_{\mathbf{C}}} R_1$ typically evaluates to a linear combination of basis vectors $\delta(r)$ with integer coefficients:

Example 1. Let $\mathbf{C} = \mathbf{FinGraph}$ be the category of finite directed multigraphs, with $\varnothing$ the empty graph. Then with $\odot \equiv \odot_{\mathscr{R}_{\mathbf{C}}}$, we find for example

$$\delta(\varnothing \hookleftarrow \varnothing \hookrightarrow \bullet) \odot \delta(\bullet \, \bullet \hookleftarrow \varnothing \hookrightarrow \varnothing)$$
$$= \sum_{\substack{\mathbf{m} \in \{(\bullet \hookleftarrow \varnothing \hookrightarrow \bullet \, \bullet), (\bullet \hookleftarrow \bullet \hookrightarrow \bullet \, \bullet), \\ (\bullet \hookleftarrow \bullet \hookrightarrow \bullet \, \bullet)\}}} \delta\left((\varnothing \hookleftarrow \varnothing \hookrightarrow \bullet) \overset{\mathbf{m}}{\triangleleft} (\bullet \, \bullet \hookleftarrow \varnothing \hookrightarrow \varnothing) \right) \tag{13}$$
$$= \delta(\bullet \, \bullet \hookleftarrow \varnothing \hookrightarrow \bullet) + 2\delta(\bullet \hookleftarrow \varnothing \hookrightarrow \varnothing).$$

The result of the composition thus captures the combinatorial insight that there are two contributions that evaluate to an isomorphic rule algebra element. More generally, one finds the following structure of compositions of rule algebra elements based upon "discrete" graph rewriting rules: letting $\bullet^{\uplus n}$ denote the n-vertex graph without edges (for $n \geq 0$), one finds (for $p, q, r, s \geq 0$)

$$\delta(\bullet^{\uplus p} \hookleftarrow \varnothing \hookrightarrow \bullet^{\uplus q}) \odot \delta(\bullet^{\uplus r} \hookleftarrow \varnothing \hookrightarrow \bullet^{\uplus s})$$
$$= \sum_{k=0}^{min(q,r)} k! \binom{q}{k} \binom{r}{k} \delta(\bullet^{\uplus (p+r-k)} \hookleftarrow \varnothing \hookrightarrow \bullet^{\uplus (q+s-k)}). \tag{14}$$

This result is further interpreted in Example 2.

Theorem 4 (Properties of $\mathscr{R}_{\mathbf{C}}^{sq}$). *For every category* **C** *satisfying Assumption 2, the associated SqPO-type rule algebra* $\mathscr{R}_{\mathbf{C}}^{sq} \equiv (\mathscr{R}_{\mathbf{C}}, \odot_{\mathscr{R}_{\mathbf{C}}})$ *is an* associative unital algebra*, with unit element* $R_\varnothing := (\varnothing \Leftarrow \varnothing)$. *(Proof: Appendix B.3)*

For the unital and associative SqPO-type rule algebras, one may provide a notion of *representations* in analogy to the DPO-type case (compare [3, 7]):

Definition 8 (Canonical representation of $\mathscr{R}_{\mathbf{C}}^{sq}$). Let **C** be a category satisfying Assumption 2, with a strict initial object $\varnothing \in ob(\mathbf{C})$, and let $\mathscr{R}_{\mathbf{C}}^{sq}$ be its associated rule algebra of SqPO type. Denote by $\hat{\mathbf{C}}$ the free $\mathbb{R}$-vector space spanned by basis vectors $|X\rangle$ indexed by isomorphism classes of objects,

$$\hat{\mathbf{C}} := span_{\mathbb{R}} \left(\{ |X\rangle \mid X \in \mathsf{obj}(\mathbf{C})_{\cong} \} \right) \equiv (\hat{C}, +, \cdot). \tag{15}$$

Then the *canonical representation* $\rho_{\mathbf{C}}^{sq} : \mathscr{R}_{\mathbf{C}}^{sq} \to End_{\mathbb{R}}(\hat{\mathbf{C}})$ of $\mathscr{R}_{\mathbf{C}}^{sq}$ is defined as a morphism from the SqPO-type rule algebra $\mathscr{R}_{\mathbf{C}}^{sq}$ to endomorphisms of $\hat{\mathbf{C}}$, with

$$\rho_{\mathbf{C}}^{sq}(\delta(p)) |X\rangle := \begin{cases} \sum_{m \in \mathsf{M}_p^{sq}(X)} |p_m(X)\rangle & \text{if } \mathsf{M}_p^{sq}(X) \neq \emptyset \\ 0_{\hat{C}} & \text{otherwise,} \end{cases} \tag{16}$$

and extended to arbitrary elements of $\mathscr{R}_{\mathbf{C}}^{sq}$ and of $\hat{\mathbf{C}}$ by linearity.

Example 2. Extending Example 1, letting $\rho \equiv \rho_{\mathbf{FinGraph}}^{sq}$, note first that by definition for all (isomorphism classes of) finite multigraphs $G \in \mathrm{obj}(\mathbf{FinGraph})_{\cong}$, $|G\rangle = \rho(\delta(G \hookleftarrow \varnothing \hookrightarrow \varnothing))|\varnothing\rangle$. With

$$\hat{D} := \rho(\delta(\varnothing \hookleftarrow \varnothing \hookrightarrow \bullet)) , \ \hat{X} := \rho(\delta(\bullet \hookleftarrow \varnothing \hookrightarrow \varnothing)) , \ |n\rangle := \left|\bullet^{\uplus n}\right\rangle (n \geq 0) , \tag{17}$$

as a consequence of (14) of Example 1 one may verify that

$$\hat{D}|0\rangle = 0_{\widehat{\mathbf{FinGraph}}} , \ \hat{D}|n\rangle = n|n-1\rangle \ (n > 0) , \ \hat{X}|n\rangle = |n+1\rangle . \tag{18}$$

In other words, the data of (17) and (18) furnishes a representation of the famous *Heisenberg-Weyl algebra* that is of fundamental importance in combinatorics and physics (see e.g. [9, 10, 11]). An alternative such representation is given by the linear operators $\hat{x}$ (multiplication by x) and ∂_x (derivation by x) acting on the $\mathbb{R}$-vector space spanned by monomials x^n, which reproduces equations isomorphic to (17) and (18), with $\partial_x x^n = nx^{n-1}$ and $\hat{x}x^n = x^{n+1}$. However, the action of $\hat{D}$ and $\hat{X}$ is of course defined on *all* states $|G\rangle$ with $G \in \mathrm{obj}(\mathbf{FinGraph})$, so that we may e.g. compute the following "derivative of a graph":

$$\hat{D}|\bullet\!\!\rightarrow\!\!\bullet\!\!\rightarrow\!\!\bullet\!\!\rightarrow\!\!\bullet\rangle = 2|\bullet\!\!\rightarrow\!\!\bullet\!\!\rightarrow\!\!\bullet\rangle + |\bullet \quad \bullet\rangle \tag{19}$$

The following theorem states that $\rho_{\mathbf{C}}^{sq}$ as given in Definition 8 is indeed a homomorphism (and thus qualifies as a representation of $\mathscr{R}_{\mathbf{C}}^{sq}$).

Theorem 5 (SqPO-type canonical representation). *For a category $\mathbf{C}$ satisfying Assumption 2, $\rho_{\mathbf{C}}^{sq} : \mathscr{R}_{\mathbf{C}}^{sq} \to End(\hat{\mathbf{C}})$ of Definition 8 is a homomorphism of unital associative algebras. (Proof: Appendix B.4)*

5 Applications of SqPO-type rule algebras to stochastic mechanics

In practical applications of stochastic rewriting systems, the type of rewriting semantics presents one of the key design choices. For example, if in a given situation a stochastic graph rewriting system should be implemented, choosing DPO- vs. SqPO-type rewriting entails two entirely different semantics in terms of the behavior of vertex deletion rules: in the former case, vertices may only be deleted if also all its incident edges are explicitly deleted as well, while in the latter case no such restriction applies (i.e. an application of a vertex deletion rule "automatically" leads to the deletion of all incident edges). Evidently, such fundamentally different behavior at the level of rewriting rules will also have strong influence on the dynamical behavior of the associated stochastic rewriting systems, whence it is of considerable practical interest to have a universal implementation of such systems available in both formalisms. We begin by specializing the general definition of continuous-time Markov chains (see e.g. [41, 1]) to the setting of SqPO-type rewriting systems in close analogy to [3, 6, 7].

Definition 9 (Continuous-time Markov Chains (CTMCs); compare [7], Def. 7.1)**.** Let $\mathbf{C}$ be a category satisfying Assumption 2, and which in addition possesses a *countable* set of isomorphism classes of objects $\mathrm{obj}(\mathbf{C})_{\cong}$. Let $\hat{\mathbf{C}}$ denote the free $\mathbb{R}$-vector space introduced in Definition 8. We define the space *Prob*($\mathbf{C}$) as the *space of sub-probability distributions* in the following sense:

$$Prob(\mathbf{C}) := \left\{ |\Psi\rangle = \sum_{o \in \mathrm{obj}(\mathbf{C})_{\cong}} \psi_o |o\rangle \ \middle| \ \forall o \in \mathrm{obj}(\mathbf{C})_{\cong} : \psi_o \in \mathbb{R}_{\geq 0} \wedge \sum_{o \in \mathrm{obj}(\mathbf{C})_{\cong}} \psi_o \leq 1 \right\} \tag{20}$$

Let *Stoch*($\mathbf{C}$) := $End_{\mathbb{R}}(Prob(\mathbf{C}))$ be the space of endomorphisms of *Prob*($\mathbf{C}$), with elements referred to as *sub-stochastic operators*. Then a ***continuous-time Markov chain (CTMC)*** is specified in terms of

a tuple of data $(|\Psi(0)\rangle, H)$, where $|\Psi(0)\rangle \in Prob(\mathbf{C})$ is the *initial state*, and where $H \in End_{\mathbb{R}}(\mathscr{S}_{\mathbf{C}})$ is the *infinitesimal generator* or *Hamiltonian* of the CTMC (with $\mathscr{S}_{\mathbf{C}}$ the space of real-valued sequences indexed by elements of $\text{obj}(\mathbf{C})_{\cong}$ and with finite coefficients). H is required to be an infinitesimal (sub-) stochastic operator, which entails that for $H \equiv (h_{o,o'})_{o,o' \in \text{obj}(\mathbf{C})_{\cong}}$ and for all $o, o' \in \text{obj}(\mathbf{C})_{\cong}$,

$$(i)\ h_{o,o} \leq 0\,,\ (ii) \forall o \neq o' : h_{o,o'} \geq 0\,,\ (iii) \sum_{o'} h_{o,o'} = 0\,. \tag{21}$$

Then this data encodes the *evolution semi-group* $\mathscr{E} : \mathbb{R}_{\geq 0} \to Stoch(\mathbf{C})$ as the (point-wise minimal non-negative) solution of the *Kolmogorov backwards* or *master equation*:

$$\tfrac{d}{dt}\mathscr{E}(t) = H\mathscr{E}(t)\,,\ \mathscr{E}(0) = \mathbb{1}_{Stoch(\mathbf{C})} \Rightarrow \forall t,t' \in \mathbb{R}_{\geq 0} : \mathscr{E}(t)\mathscr{E}(t') = \mathscr{E}(t+t') \tag{22}$$

Consequently, the *time-dependent state* $|\Psi(t)\rangle$ of the system is given by

$$\forall t \in \mathbb{R}_{\geq 0} : \quad |\Psi(t)\rangle = \mathscr{E}(t)|\Psi(0)\rangle\,. \tag{23}$$

An important technical aspect of the above definition of CTMCs is the definition of the relevant space of (sub-)probability distributions in interaction with the definition of the infinitesimal generator H and of the space $\mathscr{S}_{\mathbf{C}}$. Some remarks on this interaction and a short explanation of the relevant mathematical concepts are provided in Appendix B.5.

Our main approach in studying CTMCs based on rewriting systems will consist in analyzing the dynamical statistical behavior of so-called observables:

Definition 10 (Observables; [7], Def. 7.1). Let $\mathscr{O}_{\mathbf{C}} \subset End_{\mathbb{R}}(S_{\mathbf{C}})$ denote the space of *observables*, defined as the space of *diagonal operators*[5],

$$\mathscr{O}_{\mathbf{C}} := \{O \in End_{\mathbb{R}}(S_{\mathbf{C}}) \mid \forall X \in \text{obj}(\mathbf{C})_{\cong} : O|X\rangle = \omega_O(X)|X\rangle\,,\ \omega_O(X) \in \mathbb{R}\}\,. \tag{24}$$

We furthermore define the so-called *projection operation* $\langle| : S_{\mathbf{C}} \to \mathbb{R}$ via extending by linearity the definition of $\langle|$ acting on basis vectors of $\hat{\mathbf{C}}$,

$$\forall X \in \text{obj}(\mathbf{C})_{\cong} : \quad \langle|X\rangle := 1_{\mathbb{R}}\,. \tag{25}$$

These definitions induce a notion of *correlators* of observables (also referred to as (mixed) moments), defined for $O_1, \ldots, O_n \in \mathscr{O}_{\mathbf{C}}$ and $|\Psi\rangle \in Prob(\mathbf{C})$ as

$$\langle O_1, \ldots, O_n\rangle_{|\Psi\rangle} := \langle|O_1, \ldots, O_n|\Psi\rangle = \sum_{X \in \text{obj}(\mathbf{C})_{\cong}} \psi_X \cdot \omega_{O_1}(X) \cdots \omega_{O_n}(X)\,. \tag{26}$$

The precise relationship between the notions of CTMCs and SqPO-type rewriting rules as encoded in the corresponding SqPO-type rule algebra formalism is established in the form of the following theorem, where in particular the notion of observables is quite different in nature to the DPO-type analogon (compare Thm. 7.12 of [7]). This result is the first-of-its-kind *universal* definition of SqPO-type stochastic rewriting systems with "mass-action semantics" (where activities of productions are proportional to their number of admissible matches in a given system state).

---

[5]Depending on the concrete case, the eigenvalue $\omega_O(X)$ in $O|X\rangle = \omega_O(X)|X\rangle$ may e.g. coincide with the number of occurrences of a pattern in the object X (see also Appendix B.5).

Theorem 6 (SqPO-type stochastic mechanics framework). *Let* **C** *be a category satisfying Assumption 2. Let* $\{(O_j \overset{p_j}{\Longleftarrow} I_j) \in \mathscr{R}^{sq}_{\mathbf{C}}\}_{j \in \mathscr{J}}$ *be a (finite) set of rule algebra elements, and* $\{\kappa_j \in \mathbb{R}_{\geq 0}\}_{j \in \mathscr{J}}$ *a collection of non-zero parameters (called* base rates*). Then one may construct the Hamiltonian H of the associated CTMC from this data according to*

$$H := \hat{H} + \bar{H}, \quad \hat{H} := \sum_{j \in \mathscr{J}} \kappa_j \cdot \rho^{sq}_{\mathbf{C}} \left(O_j \overset{p_j}{\Longleftarrow} I_j \right), \quad \bar{H} := - \sum_{j \in \mathscr{J}} \kappa_j \cdot \mathbb{O}^{sq}_{I_j}. \tag{27}$$

Here, the notation $\mathbb{O}^{sq}_M$ *for arbitrary objects* $M \in \mathsf{obj}(\mathbf{C})$*) denotes the* observables *(sometimes referred to as* motif counting observables*) for the resulting CTMC of SqPO-type, with*

$$\mathbb{O}^{sq}_M := \rho^{sq}_{\mathbf{C}} \left(\delta \left(M \overset{id_M}{\longleftarrow} M \overset{id_M}{\longrightarrow} M \right) \right). \tag{28}$$

We furthermore have the SqPO-type jump-closure property*, whereby for all* $(O \overset{p}{\Longleftarrow} I) \in \mathscr{R}^{sq}_{\mathbf{C}}$

$$\langle | \rho^{sq}_{\mathbf{C}}(O \overset{p}{\Longleftarrow} I) = \langle | \mathbb{O}^{sq}_I. \tag{29}$$

Proof. See Appendix B.5.   □

6   Application example: a dynamical random graph model

In order to illustrate our novel SqPO-type stochastic mechanics framework, let us consider a dynamical system evolving on the space of finite directed multigraphs.

Example 3. Let **FinGraph** be the finitary restriction of the category **Graph** (see also Lemma 4), and denote by $\varnothing \in$ **FinGraph** the strict initial object (the empty graph). We define a stochastic SqPO rewriting system based upon rules encoding *vertex creation/deletion* ($v_\pm$) and *edge creation/deletion* ($e_\pm$):

$$
\begin{aligned}
v_+ &:= (\bullet \leftarrow \varnothing \rightarrow \varnothing) & v_- &:= (\varnothing \leftarrow \varnothing \rightarrow \bullet) \\
e_+ &:= (\bullet\!\!\rightarrow\!\!\bullet \leftarrow \bullet \ \ \bullet \rightarrow \bullet \ \ \bullet) & e_- &:= (\bullet \ \ \bullet \leftarrow \bullet \ \ \bullet \rightarrow \bullet\!\!\rightarrow\!\!\bullet)
\end{aligned} \tag{30}
$$

Together with a choice of *base rates* $v_\pm, \varepsilon_\pm \in \mathbb{R}_{\geq 0}$ and an initial state $|\Psi(0)\rangle \in Prob(\mathbf{FinGraph})$, this data defines a stochastic rewriting system with Hamiltonian $H := \hat{H} + \bar{H}$,

$$
\begin{aligned}
\hat{H} &= v_+ V_+ + v_- V_- + \varepsilon_+ E_+ + \varepsilon_- E_- \\
\bar{H} &= -v_+ \mathbb{O}_\varnothing - v_- \mathbb{O}_\bullet - \varepsilon_+ \mathbb{O}_{\bullet\,\bullet} - \varepsilon_- \mathbb{O}_{\bullet\rightarrow\bullet},
\end{aligned} \tag{31}
$$

where $V_\pm := \rho^{sq}_{\mathbf{FinGraph}}(\delta(v_\pm))$ and $E_\pm := \rho^{sq}_{\mathbf{FinGraph}}(\delta(e_\pm))$.

Despite the apparent simplicity of this model (which might be seen as a paradigmatic example of a *random graph model*), the explicit analysis via the stochastic mechanics framework will uncover a highly non-trivial interaction of the dynamics of the vertex- and of the edge-counting observables. Intuitively, since in SqPO-rewriting no conditions are posed upon vertices that are to be deleted, the model is expected to possess a vertex dynamics that is the one of a so-called *birth-death process*. If it were not for the vertex deletions, one would find a similar dynamics for the edge-counting observables (compare e.g. the DPO-type rewriting model considered in [7]). However, since deletion of vertices deletes all incident edges, the dynamics of the edge-counting observable is rendered considerably more complicated, and in

particular much less evident to foresee by heuristic arguments.

In order to compute the dynamics of the vertex counting observable $O_V := \mathbb{O}_\bullet$, we follow the approach of *exponential moment-generating functions* put forward in [3, 6, 4] and define

$$M_V(t;\lambda) := \langle| e^{\lambda O_V} |\Psi(t)\rangle, \tag{32}$$

with λ a formal variable. $M_V(t;\lambda)$ encodes the moments of the observable O_V, in that taking the *n*-th derivative of $M_V(t;\lambda)$ w.r.t. λ followed by setting $\lambda \to 0$ yields the *n*-th moment of O_V. Note that we must assume the *finiteness* of all statistical moments as standard in the probability theory literature in order for $M_V(t;\lambda)$ to be well-posed, a property that we will in the case at hand indeed derive explicitly. Referring the interested readers to [8] for further details, suffice it here to recall the following variant of the BCH formula (see e.g. [28], Prop. 3.35), for λ a formal variable and A, B two composable linear operators,

$$e^{\lambda A} B e^{-\lambda A} = e^{ad_{\lambda A}} B = \sum_{n \geq 0} \frac{\lambda^n}{n!} ad_A^{\circ n}(B), \quad ad_A(B) := AB - BA \equiv [A, B], \tag{33}$$

with the convention that $ad_A^{\circ 0}(B) := B$. The operation $[.,.]$ is typically referred to as the *commutator*. We may then derive the *formal evolution equation* for $M_V(t;\lambda)$:

$$\frac{\partial}{\partial t} M_V(t;\lambda) = \langle| e^{\lambda O_V} H |\Psi(t)\rangle = \langle| \left(e^{\lambda O_V} H e^{-\lambda O_V} \right) e^{\lambda O_V} |\Psi(t)\rangle$$
$$= \langle| \left(e^{ad_{\lambda O_V}} H \right) e^{\lambda O_V} |\Psi(t)\rangle. \tag{34}$$

Since by definition $\langle| H = 0$, it remains to compute the adjoint action $ad_{O_V}(H)$ of O_V on H:

$$ad_{O_V}(H) = v_+[O_V, V_+] + v_-[O_V, V_-] + \varepsilon_+[O_V, E_+] + \varepsilon_-[O_V, E_-]$$
$$= v_+ V_+ - v_- V_- \tag{35}$$

Here, the result that $[O_V, E_\pm] = 0$ has a very simple intuitive meaning: in applications of the linear rules $e_\pm$, the number of vertices remains unchanged, whence the vanishing of the commutator. Combining these results with the SqPO-type *jump-closure property* (cf. Theorem 6), we finally arrive at the following *formal evolution equation* for $M_V(t;\lambda)$:

$$\frac{\partial}{\partial t} M_V(t;\lambda) = v_+ \left(e^\lambda - 1 \right) \langle| V_+ e^{\lambda O_V} |\Psi(t)\rangle + v_- \left(e^{-\lambda} - 1 \right) \langle| V_- e^{\lambda O_V} |\Psi(t)\rangle$$
$$\stackrel{(29)}{=} v_+ \left(e^\lambda - 1 \right) \langle| e^{\lambda O_V} |\Psi(t)\rangle + v_- \left(e^{-\lambda} - 1 \right) \langle| O_V e^{\lambda O_V} |\Psi(t)\rangle \tag{36}$$
$$= \left(v_+ \left(e^\lambda - 1 \right) + v_- \left(e^{-\lambda} - 1 \right) \frac{\partial}{\partial \lambda} \right) M_V(t;\lambda).$$

Supposing for simplicity an initial state $|\Psi(0)\rangle = |G_0\rangle$ (for $G_0 \in \mathrm{obj}(\mathbf{Graph}_{fin})$ some graph with N_V vertices and N_E edges), we find that $M_V(0;\lambda) = \exp(\lambda N_V)$. The resulting initial value problem may be solved in closed-form via *semi-linear normal-ordering* techniques known from the combinatorics literature [22, 9, 11, 6] (see also [8, 4]), and we obtain (for $t \geq 0$)

$$M_V(t;\lambda) = \exp\left(\frac{v_+}{v_-} (e^\lambda - 1)(1 - e^{-v_- t}) \right) \left(1 + (e^\lambda - 1) e^{-v_- t} \right)^{N_V}. \tag{37}$$

In the limit $t \to \infty$, the moment-generating function becomes that of a *Poisson-distribution* (of parameter v_+/v_-), thus confirming the aforementioned intuition that the vertex-counting observable has the

dynamical behavior of a so-called *birth-death process* (see e.g. [6]).

Let us consider next the dynamics of the edge-counting observable $O_E := \mathbb{0}_{\bullet\!\rightarrow\!\bullet}$, where for brevity we will only consider the evolution of the mean edge count. The calculation of the evolution equation for the expectation value of O_E simplifies to the analogue of the so-called *Ehrenfest equation*,

$$\frac{\partial}{\partial t} \langle | O_E | \Psi(t) \rangle = \langle | O_E H | \Psi(t) \rangle = \langle | (H O_E + [O_E, H]) | \Psi(t) \rangle . \tag{38}$$

Recalling that $\langle | H = 0$, it remains to compute the commutator $[O_E, H]$:

$$
\begin{aligned}
[O_E, H] &= v_+ [O_E, V_+] + v_- [O_E, V_-] + \varepsilon_+ [O_E, E_+] + \varepsilon_- [O_E, V_-] \\
&= v_+ \cdot 0 - v_- (E_-^{0,1} + E_-^{1,0}) + \varepsilon_+ E_+ - \varepsilon_- E_- \\
E_-^{0,1} &= \rho_{\textbf{FinGraph}}^{sq} \left(\delta \left(\underset{b}{\bullet} \leftarrow \underset{b}{\bullet} \rightarrow \underset{a}{\bullet}\!\!\rightarrow\!\!\underset{b}{\bullet} \right) \right) \\
E_-^{1,0} &= \rho_{\textbf{FinGraph}}^{sq} \left(\delta \left(\underset{a}{\bullet} \leftarrow \underset{a}{\bullet} \rightarrow \underset{a}{\bullet}\!\!\rightarrow\!\!\underset{b}{\bullet} \right) \right) .
\end{aligned}
\tag{39}
$$

This calculation is a representative example of various effects that may occur in rule-algebraic commutation relations: we find a zero commutator $[O_E, V_+]$, indicating the fact that application of the vertex creation rule V_+ does not influence the edge count. The commutators $[O_E, E_\pm] = \pm E_\pm$ encode that application of the edge creation/deletion rules leads to positive/negative contributions to the edge count. Finally, the contribution of the commutator $[O_E, V_-] = -E_-^{0,1} - E_-^{1,0}$ is given by the representations of two rule algebra elements not originally present in the Hamiltonian H, with the structure of the underlying linear rules indicated by the labels a and b on the vertices (as customary in the rewriting literature). It then remains to apply the jump-closure property (Theorem 6) together with the identity $\mathbb{0}_{\bullet\bullet} = O_V(O_V - 1)$ in order to obtain the *evolution equation*

$$\frac{\partial}{\partial t} \langle | O_E | \Psi(t) \rangle = \varepsilon_+ \langle | O_V(O_V - 1) | \Psi(t) \rangle - (\varepsilon_- + 2v_-) \langle | O_E | \Psi(t) \rangle . \tag{40}$$

Together with an initial condition such as e.g. $|\Psi(0)\rangle = |G_0\rangle$ for some (finite) directed graph G_0 with N_V vertices and N_E edges, and computing the closed-form expression for the first contribution in (40) from our previous solution (37) (as $\partial_\lambda (\partial_\lambda - 1) M_V(t; \lambda)$ followed by setting $\lambda \to 0$), the initial value problem for the mean edge count evolution may be easily solved in closed form via the use of a computer algebra software such as MAPLE, MATHEMATICA or SAGE. It is also straightforward to verify that for an arbitrary initial state $|\Psi(0)\rangle = |G_0\rangle$, the limit value of the mean edge count for $t \to \infty$ reads

$$\lim_{t \to \infty} \langle | O_E | \Psi(t) \rangle = \frac{v_+^2 \varepsilon_+}{v_-^2 (2v_- + \varepsilon_-)} . \tag{41}$$

Since the rates $v_\pm$ and $\varepsilon_\pm$ are free parameters, the above result entails that in this model one may freely adjust the limit value of the average vertex count as encoded in (36) (whence v_+/v_-) as well as the limit value of the average edge count via suitable choices of the parameters $\varepsilon_\pm$. For illustration, we present some plots of the mean edge count evolution for the case $|\Psi(0)\rangle = |\varnothing\rangle$ and various choices of parameters in Figure 1.

7 Conclusion and Outlook

Extending our previous work on Double-Pushout (DPO) rewriting theories as presented in [3, 7, 4] to the important alternative setting of Sesqui-Pushout (SqPO) rewriting, we provide a number of original
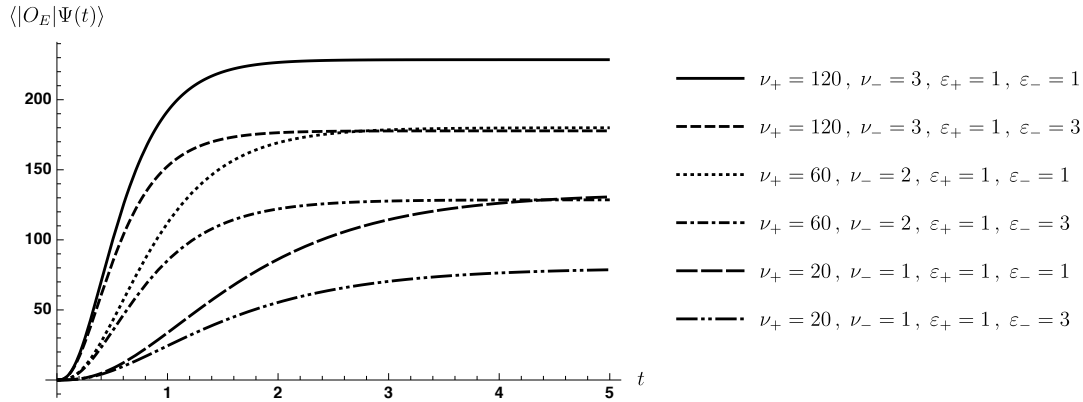
Figure 1: Time-evolution of $\langle\,|\,O_E\,|\Psi(t)\rangle$ for $|\Psi(0)\rangle = |\varnothing\rangle$.

results in the form of *concurrency* and *associativity* theorems for SqPO rewriting theories on adhesive categories. These fundamental results in turn permit us to formulate so-called *SqPO-type rule algebras*, which play a central role in our novel *universal stochastic mechanics framework*. We strongly believe that these contributions will provide fruitful grounds for further developments both in theory and practice of rewriting beyond the specialists' communities, especially in view of static analysis techniques [2].

References

[1] W.J. Anderson (1991): *Continuous-Time Markov Chains*. Springer New York, doi:10.1007/978-1-4612-3038-0.

[2] N. Behr (2019): *Tracelets and Tracelet Analysis Of Compositional Rewriting Systems*. arXiv preprint arXiv:1904.12829.

[3] N. Behr, V. Danos & I. Garnier (2016): *Stochastic mechanics of graph rewriting*. In: *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science - LICS '16*, ACM Press, doi:10.1145/2933575.2934537.

[4] N. Behr, V. Danos & I. Garnier (2019): *Combinatorial Conversion and Moment Bisimulation for Stochastic Rewriting Systems*. arXiv preprint 1904.07313.

[5] N. Behr, V. Danos, I. Garnier & T. Heindel (2016): *The algebras of graph rewriting*. arXiv:1612.06240.

[6] N. Behr, G.H.E. Duchamp & K.A. Penson (2017): *Combinatorics of Chemical Reaction Systems*. arXiv:1712.06575.

[7] N. Behr & P. Sobocinski (2018): *Rule Algebras for Adhesive Categories*. In Dan Ghica & Achim Jung, editors: *27th EACSL Annual Conference on Computer Science Logic (CSL 2018)*, *Leibniz International Proceedings in Informatics (LIPIcs)* 119, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, pp. 11:1–11:21, doi:10.4230/LIPIcs.CSL.2018.11.

[8] N. Behr & P. Sobocinski (2019): *Rule Algebras for Adhesive Categories (invited extended journal version)*. arXiv preprint 1807.00785v2.

[9] P. Blasiak, G.H.E. Duchamp, A. Horzela, K.A. Penson & A.I. Solomon (2005): *Boson normal ordering via substitutions and Sheffer-Type Polynomials*. Physics Letters A 338(2), pp. 108–116, doi:10.1016/j.physleta.2005.02.028.

[10] P. Blasiak, G.H.E. Duchamp, A. Horzela, K.A. Penson & A.I. Solomon (2010): *Combinatorial Algebra for second-quantized Quantum Theory*. Advances in Theoretical and Mathematical Physics 14(4), pp. 1209–1243, doi:10.4310/atmp.2010.v14.n4.a5.

[11] P. Blasiak & P. Flajolet (2011): *Combinatorial Models of Creation-Annihilation*. Séminaire Lotharingien de Combinatoire 65(B65c), pp. 1–78.

[12] B. Braatz, H. Ehrig, K. Gabriel & U. Golas (2010): *Finitary ℳ -Adhesive Categories*. In H. Ehrig, A. Rensink, G. Rozenberg & A. Schürr, editors: *Graph Transformations (ICGT 2010)*, Lecture Notes in Computer Science 6372, Springer, Berlin, Heidelberg, pp. 234–249, doi:10.1007/978-3-642-15928-2_16.

[13] J.R.B. Cockett & S. Lack (2003): *Restriction categories II: partial map classification*. Theoretical Computer Science 294(1-2), pp. 61–102, doi:10.1016/s0304-3975(01)00245-6.

[14] A. Corradini, D. Duval, R. Echahed, F. Prost & L. Ribeiro (2015): *AGREE – Algebraic Graph Rewriting with Controlled Embedding*. In Francesco Parisi-Presicce & Bernhard Westfechtel, editors: *Graph Transformation (ICGT 2015)*, Lecture Notes in Computer Science 9151, Springer International Publishing, Cham, pp. 35–51, doi:10.1007/978-3-319-21145-9_3.

[15] A. Corradini, D. Duval, M. Löwe, L. Ribeiro, R. Machado, A. Costa, G. Grochau Azzi, J. Santos Bezerra & L. Marques Rodrigues (2018): *On the Essence of Parallel Independence for the Double-Pushout and Sesqui-Pushout Approaches*. In R. Heckel & G. Taentzer, editors: *Graph Transformation, Specifications, and Nets*, Lecture Notes in Computer Science 10800, Springer International Publishing, Cham, pp. 1–18, doi:10.1007/978-3-319-75396-6_1.

[16] A. Corradini, T. Heindel, F. Hermann & B. König (2006): *Sesqui-Pushout Rewriting*. In A. Corradini, H. Ehrig, U. Montanari, L. Ribeiro & G. Rozenberg, editors: *Graph Transformations (ICGT 2006)*, Lecture Notes in Computer Science 4178, Springer, Berlin, Heidelberg, pp. 30–45, doi:10.1007/11841883_4.

[17] A. Corradini, U. Montanari, F. Rossi, H. Ehrig, R. Heckel & M. Löwe (1997): *Algebraic Approaches to Graph Transformation - Part I: Basic Concepts and Double Pushout Approach*. In: *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations*, World Scientific, pp. 163–246, doi:10.1142/9789812384720_0003.

[18] V. Danos, J. Feret, W. Fontana, R. Harmer, J. Hayman, J. Krivine, C. Thompson-Walsh & G. Winskel (2012): *Graphs, Rewriting and Pathway Reconstruction for Rule-Based Models*. In D. D'Souza, T. Kavitha & J. Radhakrishnan, editors: *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2012)*, Leibniz International Proceedings in Informatics (LIPIcs) 18, Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, pp. 276–288, doi:10.4230/LIPIcs.FSTTCS.2012.276.

[19] V. Danos, J. Feret, W. Fontana, R. Harmer & J. Krivine (2008): *Rule-Based Modelling, Symmetries, Refinements*. In Jasmin Fisher, editor: *Formal Methods in Systems Biology (FMSB 2008)*, Lecture Notes in Computer Science 5054, Springer, Berlin, Heidelberg, pp. 103–122, doi:10.1007/978-3-540-68413-8_8.

[20] V. Danos, J. Feret, W. Fontana, R. Harmer & J. Krivine (2010): *Abstracting the Differential Semantics of Rule-Based Models: Exact and Automated Model Reduction*. In: *2010 25th Annual IEEE Symposium on Logic in Computer Science*, IEEE, doi:10.1109/lics.2010.44.

[21] V. Danos & C. Laneve (2004): *Formal molecular biology*. Theoretical Computer Science 325(1), pp. 69–110, doi:10.1016/j.tcs.2004.03.065.

[22] G. Dattoli, P. L. Ottaviani, A. Torre & L. Vazquez (1997): *Evolution operator equations: Integration with algebraic and finite-difference methods: Applications to physical problems in classical and quantum mechanics and quantum field theory*. Riv. Nuovo Cim. 20N2, pp. 1–133, doi:10.1007/BF02907529.

[23] H. Ehrig, U. Golas, A. Habel, L. Lambers & F. Orejas (2014): *ℳ -adhesive transformation systems with nested application conditions. Part 1: parallelism, concurrency and amalgamation*. Mathematical Structures in Computer Science 24(04), doi:10.1017/s0960129512000357.

[24] H. Ehrig, A. Habel, H.J. Kreowski & F. Parisi-Presicce (1991): *Parallelism and concurrency in high-level replacement systems*. Mathematical Structures in Computer Science 1(3), pp. 361–404, doi:10.1017/s0960129500001353.

[25] H. Ehrig, A. Habel, J. Padberg & U. Prange (2004): *Adhesive High-Level Replacement Categories and Systems*. In H. Ehrig, G. Engels, F. Parisi-Presicce & G. Rozenberg, editors: *Graph Transformations (ICGT 2004)*, Lecture Notes in Computer Science 3256, Springer, Berlin, Heidelberg, pp. 144–160, doi:10.1007/978-3-540-30203-2_12.

[26] H. Ehrig, M. Pfender & H. J. Schneider (1973): *Graph-grammars: An algebraic approach*. In: *14th Annual Symposium on Switching and Automata Theory (SWAT 1973)*, IEEE, doi:10.1109/swat.1973.11.

[27] U. Golas, A. Habel & H. Ehrig (2014): *Multi-amalgamation of rules with application conditions in $\mathcal{M}$-adhesive categories*. Mathematical Structures in Computer Science 24(04), doi:10.1017/s0960129512000345.

[28] B.C. Hall (2015): *Lie Groups, Lie Algebras, and Representations*. Springer International Publishing, doi:10.1007/978-3-319-13467-3.

[29] R. Heckel (2005): *Stochastic Analysis of Graph Transformation Systems: A Case Study in P2P Networks*. In D. Van Hung & M. Wirsing, editors: *Theoretical Aspects of Computing (ICTAC 2005)*, Lecture Notes in Computer Science 3722, Springer, Berlin, Heidelberg, pp. 53–69, doi:10.1007/11560647_4.

[30] R. Heckel, H. Ehrig, U. Golas & F. Hermann (2012): *Parallelism and Concurrency of Stochastic Graph Transformations*. In H. Ehrig, G. Engels, H.J. Kreowski & G. Rozenberg, editors: *Graph Transformations (ICGT 2012)*, Lecture Notes in Computer Science 7562, Springer, Berlin, Heidelberg, pp. 96–110, doi:10.1007/978-3-642-33654-6_7.

[31] R. Heckel, G. Lajios & S. Menge (2004): *Stochastic Graph Transformation Systems*. In H. Ehrig, G. Engels, F. Parisi-Presicce & G. Rozenberg, editors: *Graph Transformations (ICGT 2004)*, Lecture Notes in Computer Science 3256, Springer, Berlin, Heidelberg, pp. 210–225, doi:10.1007/978-3-540-30203-2_16.

[32] R. Kennaway (1990): *Graph rewriting in some categories of partial morphisms*. In H. Ehrig, H.J. Kreowski & G. Rozenberg, editors: *Graph Grammars and Their Application to Computer Science (Graph Grammars 1990)*, 532, Springer, Berlin, Heidelberg, pp. 490–504, doi:10.1007/bfb0017408.

[33] C. Krause & H. Giese (2012): *Probabilistic Graph Transformation Systems*. In H. Ehrig, G. Engels, H.J. Kreowski & G. Rozenberg, editors: *Graph Transformations (ICGT 2012)*, Lecture Notes in Computer Science 7562, Springer, Berlin, Heidelberg, pp. 311–325, doi:10.1007/978-3-642-33654-6_21.

[34] S. Lack & P. Sobociński (2005): *Adhesive and quasiadhesive categories*. RAIRO - Theoretical Informatics and Applications 39(3), pp. 511–545, doi:10.1051/ita:2005028.

[35] M. Löwe (1993): *Algebraic approach to single-pushout graph transformation*. Theoretical Computer Science 109(1-2), pp. 181–224, doi:10.1016/0304-3975(93)90068-5.

[36] M. Löwe (2015): *Polymorphic Sesqui-Pushout Graph Rewriting*. In F. Parisi-Presicce & B. Westfechtel, editors: *Graph Transformation (ICGT 2015)*, Lecture Notes in Computer Science 9151, Springer International Publishing, Cham, pp. 3–18, doi:10.1007/978-3-319-21145-9_1.

[37] M. Löwe (2018): *Characterisation of Parallel Independence in AGREE-Rewriting*. In L. Lambers & J. Weber, editors: *Graph Transformation (ICGT 2018)*, Lecture Notes in Computer Science 10887, Springer International Publishing, Cham, pp. 118–133, doi:10.1007/978-3-319-92991-0_8.

[38] M. Löwe, H. König & C. Schulz (2014): *Polymorphic Single-Pushout Graph Transformation*. In S. Gnesi & A. Rensink, editors: *Fundamental Approaches to Software Engineering (FASE 2014)*, Lecture Notes in Computer Science 8411, Springer, Berlin, Heidelberg, pp. 355–369, doi:10.1007/978-3-642-54804-8_25.

[39] M. Löwe & M. Tempelmeier (2015): *Single-Pushout Rewriting of Partial Algebras*. GCM 2015 Graph Computation Models, p. 82.

[40] M. Maximova, H. Giese & C. Krause (2018): *Probabilistic timed graph transformation systems*. Journal of Logical and Algebraic Methods in Programming 101, pp. 110–131, doi:10.1016/j.jlamp.2018.09.003.

[41] J. R. Norris (1997): *Markov Chains*. Cambridge University Press, doi:10.1017/cbo9780511810633.

[42] G. Rozenberg (1997): *Handbook of Graph Grammars and Computing by Graph Transformations, Volume 1: Foundations*. World Scientific, doi:10.1142/9789812384720.

A A collection of useful technical results on adhesive categories and final pullback complements

Notational convention: Here and as throughout this paper, while evidently category-theoretical constructions such as pushouts are only unique up to isomorphisms, we will typically nevertheless pick convenient representatives of the respective isomorphism classes to simplify our notations. As standard practice in the literature, we will thus e.g. fix the convention as in (42) to choose representatives appropriately to label the pushout along an isomorphism with "equality arrows" (rather than keeping object labels generic and decorating the relevant arrow with a "$\cong$" symbol).

Lemma 2. *Let* **C** *be a category.*

(i) "Single-square" lemmata (see e.g. [8], Lem. 1.7): *In any category, given commutative diagrams of the form*

$$
\begin{array}{ccc}
A \xrightarrow{f} B & A = A & A \xrightarrow{f} B \\
\| \;\;(A)\;\; \| & \| \;\;(B)\;\; \downarrow g & \| \;\;(C)\;\; \downarrow g \;, \\
A \xrightarrow{f} B & A \xrightarrow{g} B & A \xrightarrow{g\circ f} C
\end{array}
\tag{42}
$$

(a) (A) *is a pushout for arbitrary morphisms* f,

(b) (B) *is a pullback if and only if the morphism g is a monomorphism, and*

(c) (C) *is a pullback for arbitrary morphisms* f *if g is a monomorphism.*

(ii) special adhesivity corollaries *(cf. e.g. [23], Lemma 2.6): in any adhesive category,*

(a) *pushouts along monomorphisms are also pullbacks, and*

(b) (uniqueness of pushout complements) *given a monomorphism* $A \hookrightarrow C$ *and a generic morphism* $C \to D$, *the respective pushout complement* $A \to B \xhookrightarrow{b} D$ *(if it exists) is unique up to isomorphism, and with* $b \in \mathsf{mono}(\mathbf{C})$ *(due to stability of monomorphisms under pushouts).*

(iii) "Double-square lemmata": *given commutative diagrams of the shapes*

$$
\begin{array}{ccccc}
A \xleftarrow{d} B \xleftarrow{e} C & & Z \xleftarrow{z} Z' \\
a\downarrow \;(1)\; b\downarrow \;(2)\; \downarrow c & & w\downarrow \;(3)\; \downarrow w' \\
A' \xleftarrow{d'} B' \xleftarrow{e'} C' & & Y \xleftarrow{y} Y' \\
& & v\downarrow \;(4)\; \downarrow v' \\
& & X \xleftarrow{x} X'
\end{array}
\tag{43}
$$

then in any category **C** *(cf. e.g. [34]):*

(a) Pullback-pullback (de-)composition*: If* (1) *is a pullback, then* $(1)+(2)$ *is a pullback if and only if* (2) *is a pullback.*

(b) Pushout-pushout (de-)composition*: If* (2) *is a pushout, then* $(1)+(2)$ *is a pushout if and only if* (1) *is a pushout.*

If the category is adhesive:

(c) pushout-pullback decomposition *([23], Lemma 2.6): If* $(1)+(2)$ *is a pushout,* (1) *is a pull-back, and if* $d' \in \mathsf{mono}(\mathbf{C})$ *and* $(c \in \mathsf{mono}(\mathbf{C})$ *or* $e \in \mathsf{mono}(\mathbf{C})$*), then* (1) *and* (2) *are both pushouts (and thus also pullbacks).*

(d) pullback-pushout decomposition *([27], Lem. B.2): if* $(1)+(2)$ *is a pullback,* (2) *a pushout,* (1) *commutes and* $a \in \mathsf{mono}(\mathbf{C})$*, then* (1) *is a pullback.*

(e) Horizontal FPC (de-)composition *(cf. [16], Lem. 2 and Lem. 3, compare [36], Prop. 36):*[6] *If* (1) *is an FPC (i.e. if* (d',b) *is FPC of* (a,d)*), then* $(1)+(2)$ *is an FPC if and only if* (2) *is an FPC.*

(f) Vertical FPC (de-)composition *(ibid): if* (3) *is an FPC (i.e. if* $(y.w')$ *is FPC of* (w,z)*), then*
   i. *if* (4) *is an FPC (i.e. if* (x,v') *is FPC of* (v,y)*), then* $(3)+(4)$ *is an FPC (i.e.* $(x,v' \circ w')$ *is FPC of* $(v \circ w, z)$*);*
   ii. *if* $(3)+(4)$ *is an FPC (i.e. if* $(x,v' \circ w')$ *is FPC of* $(v \circ w, z)$*), and if* (4) *is a pullback, then* (4) *is an FPC (i.e.* (x,v') *is FPC of* (v,y)*).*

(g) Vertical FPC-pullback decomposition *(compare [36], Lem. 38): If* $v \in \mathsf{mono}(\mathbf{C})$*, if* (4) *is a pullback and if* $(3)+(4)$ *is an FPC (i.e. if* $(x,v' \circ w')$ *is FPC of* $(v \circ w, z)$*), then* (3) *and* (4) *are FPCs.*

(h) Vertical FPC-pushout decomposition[7]*: If all morphisms of the squares* (3) *and* (4) *except v are in* $\mathsf{mono}(\mathbf{C})$*, if* $v \circ w \in \mathsf{mono}(\mathbf{C})$*, if* (3) *is a pushout and if* $(3)+(4)$ *is an FPC (i.e. if* $(x,v' \circ w')$ *is FPC of* $(v \circ w, z)$*), then* (4) *is an FPC and* $v \in \mathsf{mono}(\mathbf{C})$*.*

Proof. Referring to the references above for the proofs of the (well-known) statements (where necessary by specializing the more general case of $\mathcal{M}$-adhesive categories to the case of adhesive categories via setting $\mathcal{M}$ to the class of all monomorphisms), it remains to prove our novel vertical FPC-pushout decomposition result. To this end, we first invoke pullback-pushout decomposition (Lemma 2(3d)) in order to demonstrate that since $(3)+(4)$ is an FPC and thus also a pullback, and since (3) is a pushout and since $x \in \mathsf{mono}(\mathbf{C})$, (4) is a pullback. By applying vertical FPC-pullback decomposition, we may conclude that (4) is an FPC. In order to demonstrate that $v \in \mathsf{mono}(\mathbf{C})$, construct the commutative cube below left:

$$(44)$$

Since the bottom square is the FPC (and thus pullback) $(3)+(4)$, and since the right square is a pullback via Lemma 2(1c) (because $v' \in \mathsf{mono}(\mathbf{C})$), by pullback composition the square $\square(Z',Z,X,Y')$ (the right plus the bottom square) is a pullback. Thus assembling the commutative diagram as shown above right,

---

[6]It is worthwhile emphasizing that in these FPC-related lemmata, the "orientation" of the diagrams plays an important role. Moreover, the precise identity of the pair of morphisms that plays the role of the final pullback complement in a given square may be inferred from the "orientation" specified in the condition part of each statement.

[7]We invite the interested readers to compare the precise formulation of the vertical FPC-pushout decomposition result to its concrete applications in the setting of the proof of the concurrency theorem, for which it has been specifically developed.

since by assumption (3) is a pushout and all arrows except v are monomorphisms, invoking Theorem 7 below permits to prove that also $v \in \mathsf{mono}(\mathbf{C})$. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

The last part of the preceding proof relied upon one of the quintessential properties of adhesive categories in view of associative rewriting theories:

Theorem 7 (Effective unions; [34], Thm. 5.1). *In an adhesive category* $\mathbf{C}$, *given a commutative diagram such as in the middle of* (1), *if all morphisms except the morphism x are monomorphisms, if the square marked* (A) *is a pushout and if the exterior square is a pullback, then x is also a monomorphism.*

The following result provides several important facts on FPCs.

Lemma 3 (cf. [36], Fact 2, and [16], Lemma 2 and Proposition 2). *Let* $\mathbf{C}$ *be adhesive. For an arbitrary morphism* $f : A \to B$, (id_B, f) *is an FPC of* (f, id_A) *and vice versa. Moreover, every pushout square along monomorphisms is also an FPC square. FPCs are unique up to isomorphism and preserve monomorphisms. The latter property entails that if* $C \xleftarrow{d} D \xleftarrow{b} A$ *is the FPC of* $C \xleftarrow{c} B \xleftarrow{a} A$ *and if* $a \in \mathsf{mono}(\mathbf{C})$, *then also* $d \in \mathsf{mono}(\mathbf{C})$ *and vice versa (while* $c \in \mathsf{mono}(\mathbf{C})$ *entails that* $b \in \mathsf{mono}(\mathbf{C})$ *by stability of monomorphisms under pullbacks in an adhesive category* $\mathbf{C}$).

For concreteness, we quote the following explicit construction of FPCs in the category **Graph** of directed multigraphs:

Lemma 4 (FPCs for graphs; [16], Sec. 4.1 and Construction 5). *Let* **Graph** *denote the adhesive category of directed multigraphs, with*

- $\mathsf{obj}(\mathbf{Graph})$: *(multi-)graphs, i.e. tuples* $G = (V_G, E_G, \mathsf{src}_G : E_G \to V_G, \mathsf{trg}_G : E_G \to V_G)$, *with* V_G *the set of vertices,* E_G *the set of edges (with* $V_G \cap E_G = \emptyset$), src_G *the source and* trg_G *the target maps*

- $\mathsf{mor}(\mathbf{Graph})$: *graph homomorphisms* $f : G \to H$, *specified in terms of pairs of morphisms* $(f_V : V_G \to V_H, f_E : E_G \to E_H)$ *such that* $\mathsf{src}_H \circ f_E = f_V \circ \mathsf{src}_G$ *and* $\mathsf{trg}_H \circ f_E = f_V \circ \mathsf{trg}_G$.

Let $\mathsf{mono}(\mathbf{Graph})$ *denote the class of all injective graph morphisms. Then for every composable pair of monomorphisms* $K \xhookrightarrow{i} I \xhookrightarrow{m} X$, *the FPC exists and is constructed explicitly as*[8] $K \xrightarrow{m|K} \overline{K} \xhookrightarrow{\subseteq} X$, *where* $\xhookrightarrow{\subseteq}$ *denotes an inclusion morphism, and where the graph* $\overline{K}$ *reads*

$$
\begin{aligned}
V_{\overline{K}} &= V_X \setminus m[V_I \setminus V_K] \\
E_{\overline{K}} &= \{ e \in E_X \setminus m[E_I \setminus E_K] \,|\, \mathsf{src}_X(e) \in V_{\overline{K}} \wedge \mathsf{trg}_X(e) \in V_{\overline{K}} \}.
\end{aligned}
\tag{45}
$$

B   Proofs

B.1   Proof of the SqPO concurrency theorem

Proof. Throughout this proof, in each individual constructive step it may be verified that due to the stability of monomorphisms under pullbacks and pushouts, due to the various decomposition lemmata provided in the form of Lemma 2, and on occasion due to Theorem 7 on effective unions in adhesive categories, all morphisms induced in the "Synthesis" and "Analysis" steps are in fact monomorphisms. For better readability, we will not explicitly mention the individual reasoning steps on this point except for a few intricate sub-steps, since they may be recovered in a straightforward manner.

---

[8] The quoted Construction 5 of [16] is slightly more general, in that the morphism m may be permitted to not be a monomorphism; we will however have no application for such a generalization in our framework.

— **Synthesis:** Consider the setting presented in (46a). Here, we have obtained the candidate match $\mathbf{n} = (I_2 \leftarrow M_{21} \rightarrow O_1)$ via pulling back the cospan $(I_2 \rightarrow X_1 \leftarrow O_1)$. Next, we construct N_{21} via taking the pushout of $\mathbf{n}$, which induces a unique arrow $N_{21} \rightarrow X_1$ that is according to Theorem 7 a monomorphism. The diagram in (46b) is obtained by taking the pullbacks of the spans $\overline{K}_i \rightarrow X_1 \leftarrow N_{21}$ (obtaining the objects K_i', for $i = 1, 2$). By virtue of pushout-pullback decomposition (Lemma 2(3c)), the squares $\square(K_1', \overline{K}_1, X_1, N_{21})$ and $\square(K_1, K_1', N_{21}, O_1)$ are pushouts. Invoking vertical FPC-pullback decomposition (Lemma 2(3g)), the squares $\square(K_2', N_{21}, X_1, \overline{K}_2)$ and $\square(K_2, I_2, N_{21}, K_2')$ are FPCs. Next, letting $O_{21} := \mathsf{PO}(O_2 \leftarrow K_2 \rightarrow K_2')$ and $I_{21} := \mathsf{PO}(O_1 \leftarrow K_1 \rightarrow K_1')$, we have via vertical FPC-pushout decomposition (Lemma 2(3h)) that the resulting two squares on the very right (the ones involving I_{21}) are FPCs and that the arrow $I_{21} \rightarrow X_0$ is a monomorphism, while pushout-pushout decomposition (Lemma 2(3b)) entails that the two newly formed squares on the very left (the ones involving O_{21}) are pushouts.

The final step as depicted in (46c) consists in constructing $K_{21} = \mathsf{PB}(K_2' \rightarrow N_{21} \leftarrow K_1')$ and $\overline{K}_{21} = \mathsf{PB}(\overline{K}_2 \rightarrow X_1 \leftarrow \overline{K}_1)$, which by universality of pullbacks induces a unique arrow $K_{21} \rightarrow \overline{K}_{21}$. By invoking pullback-pullback decomposition (Lemma 2(3a)), one may show that the squares $\square(K_{21}, \overline{K}_{21}, \overline{K}_i, K_i')$ (for $i = 1, 2$) are pullbacks. Since the square $\square(K_1', \overline{K}_1, X_1, N_{21})$ is a pushout, via the van Kampen property (cf. Def. 1) the square $\square(K_{21}, \overline{K}_{21}, \overline{K}_2, K_2')$ is a pushout. Since according to Lemma 3 pushouts are also FPCs, it follows via horizontal composition of FPCs (Lemma 2(3e)) that the square $\square(K_{21}, \overline{K}_{21}, X_1, N_{21})$ is an FPC. Noting that the pushout square $\square(K_1', \overline{K}_1, X_1, N_{21})$ is an FPC as well, it follows via horizontal decomposition of FPCs (Lemma 2(3e)) that $\square(K_{21}, \overline{K}_{21}, \overline{K}_1, K_1')$ is an FPC. Thus the claim follows by invoking pushout composition (Lemma 2(3b)) and horizontal FPC composition (Lemma 2(3e)) in order to obtain the pushout square $\square(K_{21}, \overline{K}_{21}, X_2, O_{21})$ and the FPC square $\square(K_{21}, \overline{K}_{21}, X_0, I_{21})$.

— **Analysis:** Given the setting as depicted in (47a) of Figure 3, where the top row has the structure of an SqPO-composition (compare (4)), where the square $\square(K_{21}, K_1', N_{21}, K_2')$ is a pullback, the left "curvy" bottom square a pushout and the right "curvy" bottom square an FPC, we may obtain the configuration of (47c) as follows: construct[9] $\overline{K}_1$ via taking the final pullback complement of $K_1' \rightarrow I_{21} \rightarrow X_0$ (which implies the existence of an arrow $\overline{K}_{21} \rightarrow \overline{K}_1$ via the FPC property). Note in particular that according to Lemma 3, both arrows constructed via forming the aforementioned FPC are monomorphisms, and thus by stability of monomorphisms in an adhesive category (compare Definition 1 and Lemma 2(1c)), the arrow $\overline{K}_{21} \rightarrow \overline{K}_1$ is a monomorphism as well. Next, take the pushout $X_1 = \mathsf{PO}(\overline{K}_1 \leftarrow K_1' \rightarrow N_{21})$, followed by constructing $\overline{K}_2$ as the final pullback complement of $K_2' \rightarrow N_{21} \rightarrow X_1$ (which implies due to the FPC property of the resulting square $\square(K_2', \overline{K}_2, X_1, N_{21})$ the existence of an arrow $\overline{K}_{21} \rightarrow \overline{K}_2$). Invoking pullback-pullback decomposition (Lemma 2(3a)) twice, followed by the van Kampen property (Def. 1), we may conclude that the square $\square(K_{21}, \overline{K}_{21}, \overline{K}_2, K_2')$ is a pushout. Thus invoking pushout-pushout decomposition (Lemma 2(3b)), we find that also $\square(K_2', \overline{K}_2, X_2, O_{21})$ is a pushout. We finally arrive at the configuration in (47d) via composition of pushout and FPC squares, respectively, thus concluding the proof. $\square$

B.2   Proof of the SqPO associativity theorem

Our proof strategy will be closely related to the one presented in [7] (with full technical details provided in [8]) for the analogous associativity theorem in the DPO-rewriting case. However, the SqPO-type case poses considerable additional challenges, since this rewriting semantics yields diagrams of a rather heterogeneous nature (including pullbacks, pushouts, pushout complements and FPCs) as compared to
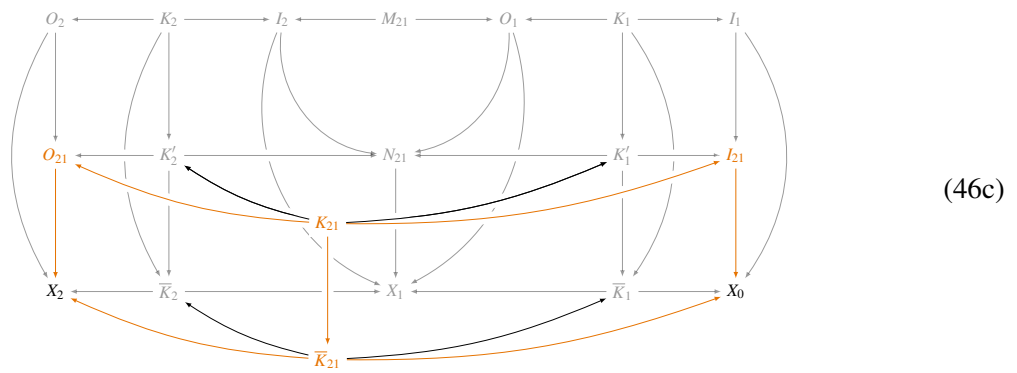
---

[9]Note that it is precisely in this step and the following step that we require the existence of FPCs for arbitrary pairs of monomorphisms as per Assumption 1.
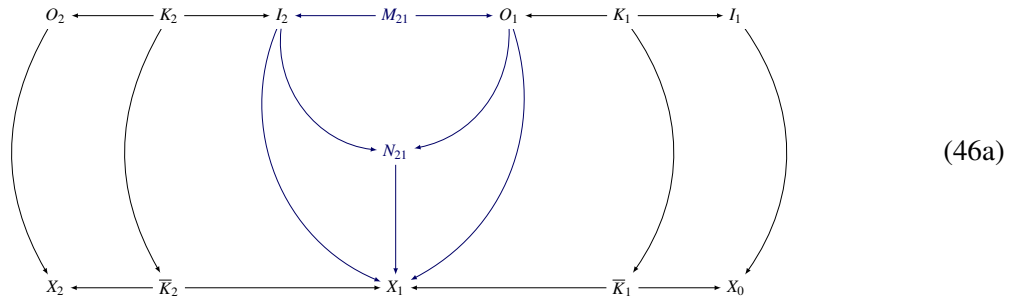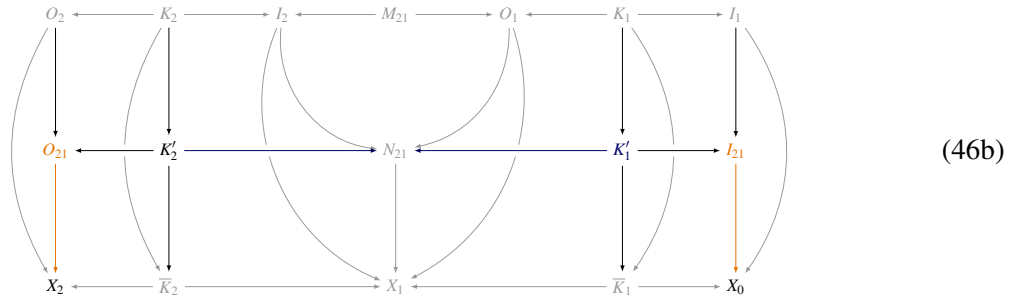
Figure 2: *Synthesis* part of the concurrency theorem.

the DPO case, and in addition unlike DPO-type rule compositions, SqPO-type compositions are not reversible in general, which necessitates an independent proof of both directions of the bijective correspondence.

Proof. We first prove the claim in the "⇒" direction, i.e. starting from the set of data

$$\mathbf{m}_{21} = (O_2 \leftarrow M_{21} \rightarrow I_1) \in \mathbf{M}_{p_2}^{sq}(p_1)$$
$$\mathbf{m}_{3(21)} = (O_3 \leftarrow M_{3(21)} \rightarrow I_{21}) \in \mathbf{M}_{p_3}^{sq}(p_{21}), \qquad p_{21} = p_2 \overset{\mathbf{m}_{21}}{\triangleleft} p_1, \tag{48}$$

we have to demonstrate that one may uniquely (up to isomorphisms) construct from this information a pair of admissible matches

$$\mathbf{m}_{32} = (O_3 \leftarrow M_{32} \rightarrow I_2) \in \mathbf{M}_{p_3}^{sq}(p_2)$$
$$\mathbf{m}_{(32)1} = (O_{32} \leftarrow M_{(32)1} \rightarrow I_1) \in \mathbf{M}_{p_{32}}^{sq}(p_1), \qquad p_{32} = p_3 \overset{\mathbf{m}_{32}}{\triangleleft} p_2, \tag{49}$$

and such that the property described in (6) holds. We begin by forming the SqPO-composite rule $p_{3(21)} = p_3 \overset{\mathbf{m}_{3(21)}}{\triangleleft} p_{21}$, which results in the diagram

$$
\tag{50}
$$

by virtue of invoking SqPO-composition twice. For the remainder of the proof, it is very important to precisely determine the nature of each of the squares in this diagram:

- To clarify the structure of the rightmost four squares at the bottom, consider the setting presented in (47a): since in the definition of the SqPO-composition as presented in (4) the nature of the squares to the right is that of pushout complement and pushout, respectively, it may be verified that applying the analysis procedure to the diagram in (47a) with thus the "curvy" front and right bottom faces both pushouts, one eventually arrives (by virtue of pushout-pushout and pushout-pullback decomposition) at the setting depicted in (47c) with all squares in the bottom row being pushouts. Thus the rightmost four squares at the bottom of (50) are all pushouts.

- By virtue of the definition of SqPO-composition, all vertical squares in the back of (47c) are pushouts, except for the square $\square(K_2, K_2', N_{21}, I_2)$, which is an FPC. Analogously, the bottom left-most three squares are (in order from left to right) a pushout, an FPC and a pushout.

Constructing the pullback $M_{32} = \mathsf{PB}(M_{3(21)} \to O_{21} \leftarrow O_2)$ (which by composition of pullbacks also leads to an arrow $M_{32} \to I_3$) and forming the three additional vertical squares on the far left in the evident fashion in the diagram below

$$
\tag{51}
$$

allows us to construct $N_{32} = \mathsf{PO}(I_3 \leftarrow M_{32} \to O_2)$, which in turn via universality of pushouts uniquely induces an arrow $N_{32} \to N_{3(21)}$:

$$
\tag{52}
$$

Here, the rightmost three squares on the top are formed in the evident fashion (and they are pushouts by virtue of Lemma 2(1a)), while the other arrows of the above diagram are constructed as follows:

$$
\begin{aligned}
K_3' &= \mathsf{PB}(\overline{K}_3 \to N_{3(21)} \leftarrow N_{32}), & O_{32} &= \mathsf{PO}(K_3' \leftarrow K_3 \to O_3) \\
K_2'' &= \mathsf{PB}(N_{32} \to N_{3(21)} \leftarrow \overline{K}_2), & I_{32} &= \mathsf{PO}(K_2'' \leftarrow K_2 \to I_2)
\end{aligned}
\tag{53}
$$

Invoking pushout-pullback, pushout-pushout and vertical FPC-pullback decompositions, it may be verified that (describing positions of front and top square pairs by the position of the respective front square, from left to right)

- the leftmost front and top squares are pushouts,

- the second from the left front and top squares are FPCs,

- the third from the left front and top squares are pushouts,

- in the next adjacent pair, the front square is an FPC and the top square a pushout,

- the second from the right front and top squares are pushouts, and

- the rightmost front and top squares are pushouts.

Defining the pullback object $M_{(32)1} = \mathsf{PB}(I_{32} \to N_{3(21)} \leftarrow O_1)$, thus inducing an arrow $M_{21} \to M_{3(21)}$,

$$(54)$$

it remains to verify that the square $\square(M_{3(21)}, I_{32}, N_{3(21)}, O_1)$ is not only a pullback, but also a pushout square. This part of the proof requires a somewhat intricate diagram chase; since the required arguments are identical[10] to the "⇒" part of proof of DPO-type associativity as presented in [8], we omit this part of the proof here in the interest of brevity.

It thus remains to prove the claim in the "⇐" direction, i.e. starting from the set of data

$$\mathbf{m}_{32} = (O_3 \leftarrow M_{32} \to I_2) \in \mathbf{M}_{p_3}^{sq}(p_2)$$

$$\mathbf{m}_{(32)1} = (O_{32} \leftarrow M_{(32)1} \to I_1) \in \mathbf{M}_{p_{32}}^{sq}(p_1), \qquad p_{32} = p_3 \overset{\mathbf{m}_{32}}{\vartriangleleft} p_2,$$

$$(55)$$

we need to demonstrate that one may uniquely (up to isomorphisms) construct from this information a pair of admissible matches

$$\mathbf{m}_{21} = (O_2 \leftarrow M_{21} \to I_1) \in \mathbf{M}_{p_2}^{sq}(p_1)$$

$$\mathbf{m}_{3(21)} = (O_3 \leftarrow M_{3(21)} \to I_{21}) \in \mathbf{M}_{p_3}^{sq}(p_{21}), \qquad p_{21} = p_2 \overset{\mathbf{m}_{21}}{\vartriangleleft} p_1,$$

$$(56)$$

and such that the property described in (6) holds. We begin by forming the SqPO-composite rule $p_{(32)1} = p_{32} \overset{\mathbf{m}_{(32)1}}{\vartriangleleft} p_1$, which results in the diagram

$$(57)$$

---

[10]More precisely, the only difference in the structure of the relevant sub-diagram compared to the DPO case resides in the two FPC squares in the front and back in fourth position from the left (which happen to be pushout squares in the corresponding DPO-type proof), but the structure of this part of the diagram is not explicitly used in the proof in the DPO variant of the theorem, whence the claim follows.

by virtue of invoking SqPO-composition twice. A careful inspection of the definition of the SqPO-composition and of the analysis part of the SqPO concurrency theorem permit to verify that the nature of all squares thus constructed coincides precisely with the nature of the corresponding squares in the "$\Rightarrow$" part of the proof.

Constructing the pullback $M_{21} = \mathsf{PB}(I_2 \to I_{32} \leftarrow M_{(32)1})$ (which by composition of pullbacks also leads to an arrow $M_{21} \to O_1$) and forming the three additional vertical squares on the far right in the evident fashion in the diagram below

(58)

allows us to construct $N_{21} = \mathsf{PO}(I_2 \leftarrow M_{21} \to O_1)$, which in turn via the universal property of pushouts induces an arrow $N_{21} \to N_{(32)1}$:

(59)

The remaining new squares of the above diagram are constructed as follows:

$$K_2' = \mathsf{PB}(\overline{K}_2 \to N_{(32)1} \leftarrow N_{21}) \quad O_{21} = \mathsf{PO}(O_2 \leftarrow K_2 \to K_2'). \tag{60}$$

Moreover, by virtue of vertical FPC composition, the square $\square(K_3, \overline{K}_3, N_{3(21)}, I_3)$ is an FPC, while via pushout composition the square $\square(K_3, \overline{K}_3, O_{321}, O_3)$ is a pushout.

Again, the nature of all squares constructed thus far coincides precisely with the structure as presented in the "$\Rightarrow$" part of the proof, with one notable exception: by virtue of vertical FPC-pullback decomposition, we may only conclude that the square $\square(K_2', \overline{K}_2, N_{(32)1}, N_{21})$ is an FPC (but at this point we do *not* know whether it is also a pushout as in the analogous part of the diagram in the "$\Rightarrow$" part of the proof). However, an auxiliary calculation demonstrates that this square is in fact a pushout in disguise — consider the following "splitting" of the relevant sub-part of the diagram as shown below:


(61)

The precise steps are as follows: the front left square in the diagram above left is an FPC; thus if one takes the pushout $N'_{(32)1} = \text{PO}(\overline{K}_2 \leftarrow K'_2 \rightarrow N_{21})$ as well as in the bottom back the pushout along the isomorphism of K''_2 as displayed (yielding the arrows in the middle), followed by taking the pullback $O'_1 = \text{PB}(N'_{(32)1}, N_{(32)1}, O_1)$ (which entails that the arrow $M_{3(21)} \rightarrow O'_1$ exists), it is straightforward to verify that $O'_1 \cong O_1$. Invoking the van Kampen property (recalling that by definition the right square on the bottom is a pushout), we find that $\square(M_{3(21)}, O'_1, N'_{(32)1}, I_{32})$ is a pushout. Thus by pushout-pushout decomposition, the square $\square(O'_1, O_1, N_{(32)1}, N'_{(32)1})$ is a pushout, whence $N'_{(32)1} \cong N_{(32)1}$. This in summary entails[11] that the square $\square(K'_2, \overline{K}_2, N_{(32)1}, N_{21})$ is not only an FPC, but in fact also a pushout.

Back to the main proof, defining the pullback object

$$M_{3(21)} = \text{PB}(I_3 \rightarrow N_{(32)1} \leftarrow O_{21}),$$

thus inducing an arrow $M_{32} \rightarrow M_{3(21)}$,

(62)
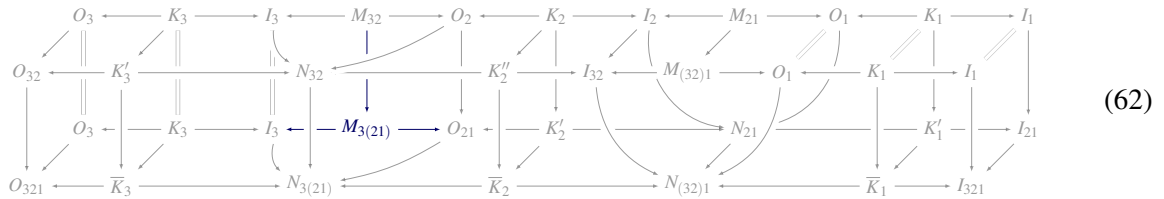
it remains to verify that the square $\square(M_{3(21)}, O_{21}, N_{3(21)}, I_3)$ is not only a pullback, but also a pushout square. Let us construct the auxiliary diagram as depicted in Figure 4, with objects obtained via taking suitable pullbacks as indicated. The four cubes that are drawn separately are the top, back, bottom and front cubes induced via the newly constructed arrows, and are oriented such that one may easily apply the van Kampen property in the next step of the proof (which in most cases requires a suitable 3d-rotation).

Invoking pullback-pullback decomposition and the van Kampen property repeatedly, it may be verified that in the relevant sub-diagram as presented below

(63)
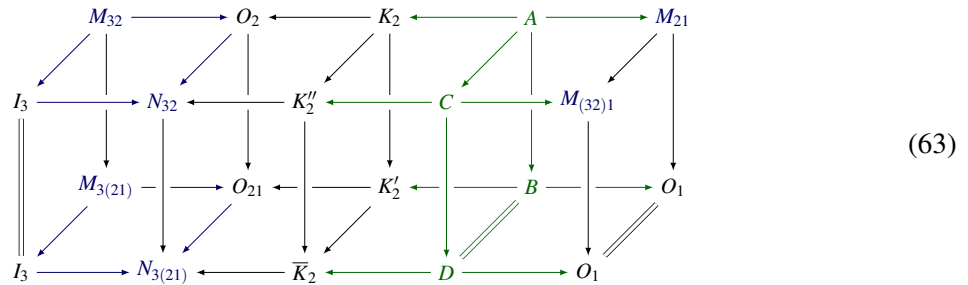
we find the following structure of the squares:

- All squares on the top are pushouts, except the second one from the right (which is a pullback).

- The second and third square from the left in the back of the diagram are pushouts, the other two back squares are pullbacks, with the same structure for the front squares.

---

[11]Coincidentally, at this point we are back into full structural analogy to the "$\Rightarrow$" part of the proof, a necessary prerequisite for completing this part of the proof as it will turn out.
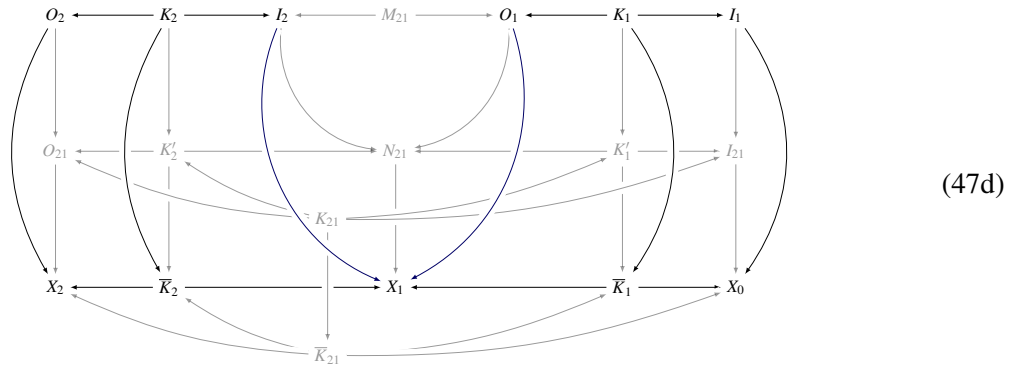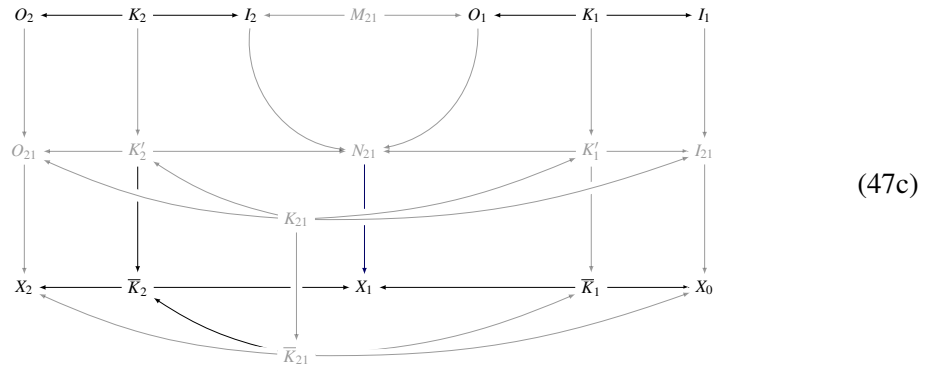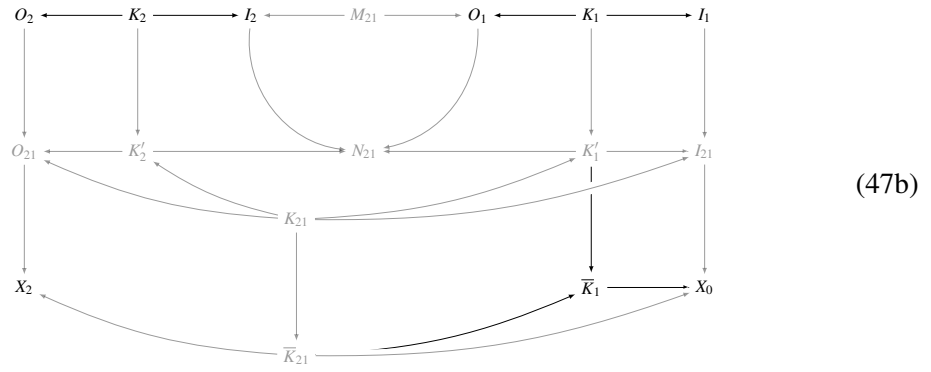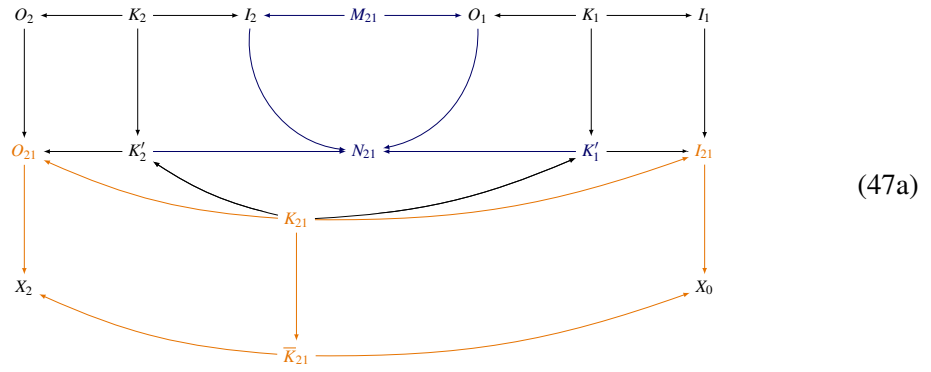
(47a)

(47b)

(47c)

(47d)

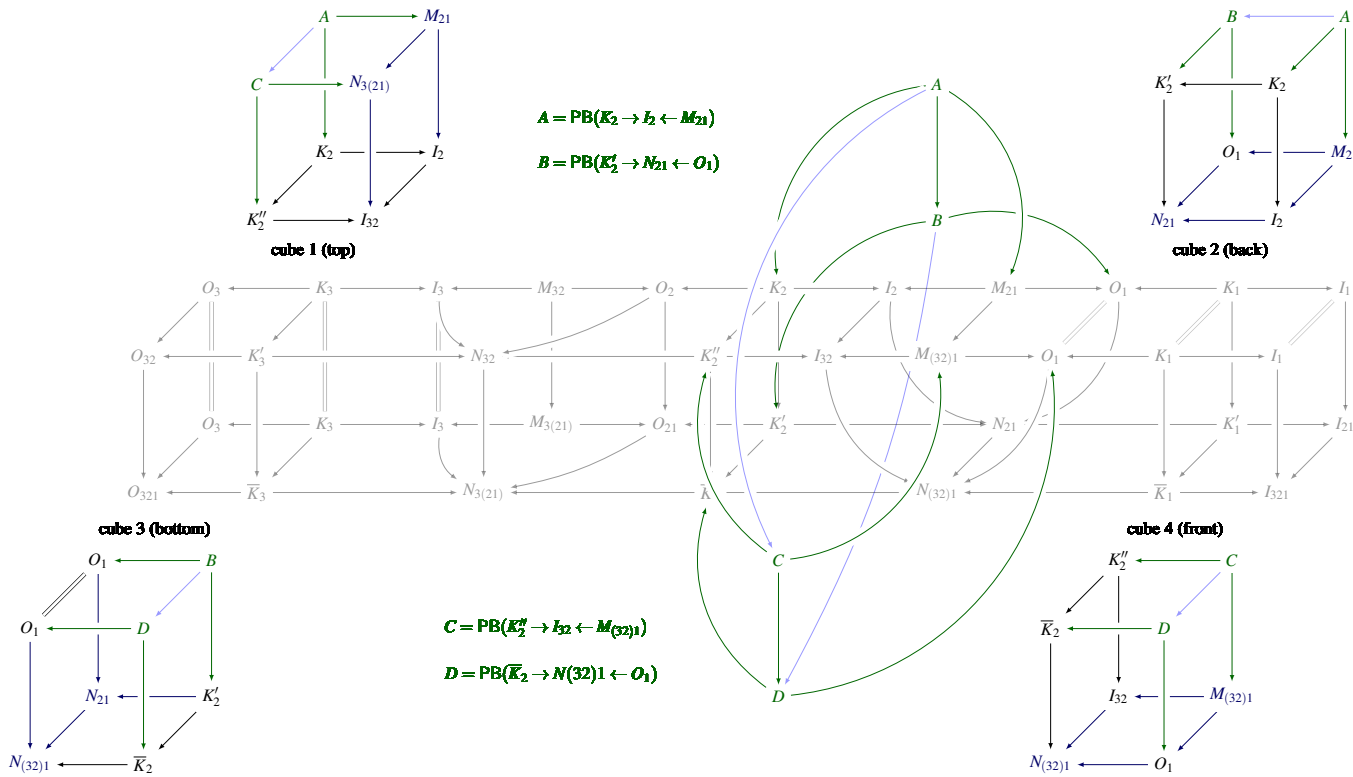Figure 3: *Analysis* part of the concurrency theorem.

Figure 4: Auxiliary diagram for the second part of the SqPO associativity proof.

- Counting from left to right, the second and fourth square on the bottom are pushouts, the other two are pullbacks.

In particular, as indicated this entails that $D \cong B$. We proceed by performing the following "splitting" of the diagram:

$$(64)$$

Start the "splitting" via taking the pushouts $N'_{32} = \mathsf{PO}(I_3 \leftarrow I_3 \rightarrow N_{32})$ (which entails that $N'_{32} \cong N_{32}$) and $O'_2 = \mathsf{PO}(M_{3(21)} \leftarrow M_{32} \rightarrow O_2)$. By pullback-pullback decomposition followed by pushout-pullback decomposition, we may conclude that the resulting square $\square(O'_2, N'_{32}, I_3, M_{3(21)})$ is a pushout. Note also that all vertical squares in the bottom left part of the diagram thus constructed are pullbacks (by virtue of suitable pullback decompositions).

Next, construct the pullbacks $\overline{K}'_2 = \mathsf{PB}(N'_{32} \rightarrow N_{3(21)} \leftarrow \overline{K}_2)$ and $K'''_2 = \mathsf{PB}(O'_2 \rightarrow O_{21} \leftarrow K'_2)$. By pushout-pullback decomposition, in the diagram on the right the top and bottom front squares and back squares in the second column are pushouts, while the square $\square(K'''_2, K'_2, \overline{K}_2, \overline{K}'_2)$ is a pullback. Performing the precise same steps in the next column, i.e. via taking the pullbacks $D' = \mathsf{PB}(\overline{K}'_2 \rightarrow \overline{K}_2 \leftarrow D)$ and $B' = \mathsf{PB}(K'''_2 \rightarrow K'_2 \leftarrow B)$, we obtain via pushout-pullback decomposition that the top and bottom front and top and bottom back squares in the third column are pushouts, while the square $\square(B', B, D, D')$ is a pullback. But since $D \cong B$ and since isomorphisms are stable under pullback, we conclude that $D' \cong B'$, and thus that the square $\square(B', B, D, D')$ is in fact a pushout.

By pushout composition, we may thus conclude that $\square(K'''_2, K'_2, \overline{K}_2, \overline{K}'_2)$ is a pushout, whence the square $\square(O'_2, O_{21}, N_{3(21)}, N'_{32})$ is a pushout, which finally allows us to verify that $\square(M_{3(21)}, O_{21}, N_{3(21)}, I_3)$ is a pushout. This concludes the proof of the SqPO-type associativity theorem.   □

B.3   Proof of the theorem on SqPO rule algebra properties

Proof. Associativity of $\odot_{\mathscr{R}_{\mathbf{C}}}$ follows from the associativity of the operation $. \mathbin{\dot{\lhd}} .$ proved in Theorem 3. The claim that $R_\varnothing = \delta(\varnothing \leftarrow \varnothing \rightarrow \varnothing)$ is the unit element of the rule algebra $\mathscr{R}^{sq}_{\mathbf{C}}$ follows directly from the definition of the rule algebra product for $R_\varnothing \odot_{\mathscr{R}_{\mathbf{C}}} R$ and $R \odot_{\mathscr{R}_{\mathbf{C}}} R_\varnothing$ for $R \in \mathscr{R}^{sq}_{\mathbf{C}}$. More concretely, we present below the category-theoretic composition calculation that underlies the equation $R_\varnothing \odot_{\mathscr{R}_{\mathbf{C}}} R = R$:

$$(65)$$

Here, it is important to note that the pushout complement used to construct the square marked POC always exists (see Lemma 2(1a)), whence the claim follows. $\square$

B.4 Proof of the SqPO canonical representation theorem

Proof. In order for $\rho_{\mathbf{C}}^{sq}$ to qualify as an algebra homomorphism (of unital associative algebras $\mathscr{R}_{\mathbf{C}}^{sq}$ and $End(\hat{\mathbf{C}})$), we must have (with $R_\varnothing = \delta(p_\varnothing)$, $p_\varnothing = (\varnothing \leftarrow \varnothing \rightarrow \varnothing)$)

$$(i)\ \rho_{\mathbf{C}}^{sq}(R_\varnothing) = \mathbb{1}_{End(\hat{\mathbf{C}})}$$
$$(ii)\ \forall R_1, R_2 \in \mathscr{R}_{\mathbf{C}}^{sq}:\ \rho_{\mathbf{C}}^{sq}(R_1 \odot_{\mathscr{R}_{\mathbf{C}}} R_2) = \rho_{\mathbf{C}}^{sq}(R_1)\rho_{\mathbf{C}}^{Sq}(R_1)\,.$$

Due to linearity, it suffices to prove the two properties on basis elements $\delta(p), \delta(q)$ of $\mathscr{R}_{\mathbf{C}}^{sq}$ (for $p, q \in Lin(\mathbf{C})$) and on basis elements $|X\rangle$ of $\hat{\mathbf{C}}$. Property (i) follows directly from the definition,

$$\forall X \in \mathrm{obj}(\mathbf{C})_\cong:\quad \rho_{\mathbf{C}}^{sq}(R_\varnothing)|X\rangle \overset{(16)}{=} \sum_{m \in \mathsf{M}_{r_\varnothing}^{sq}(X)} |(r_\varnothing)_m(X)\rangle = |X\rangle\,.$$

Property (ii) follows from Theorem 2 (the SqPO-type concurrency theorem): for all basis elements $\delta(p), \delta(q) \in \mathscr{R}_{\mathbf{C}}^{sq}$ (with $p, q \in Lin(\mathbf{C})$) and for all $X \in \mathrm{obj}(\mathbf{C})$,

$$\begin{aligned}
\rho_{\mathbf{C}}^{sq}(\delta(q) \odot_{\mathbf{C}} \delta(p))|X\rangle &\overset{(11)}{=} \sum_{\mathbf{d} \in \mathbf{M}_q^{sq}(p)} \rho_{\mathbf{C}}^{sq}\left(\delta\left(q \overset{\mathbf{d}}{\lhd} p\right)\right)|X\rangle \\
&\overset{(16)}{=} \sum_{\mathbf{d} \in \mathbf{M}_q^{sq}(p)} \sum_{e \in \mathsf{M}_{r_\mathbf{d}}^{sq}(X)} |(r_\mathbf{d})_e(X)\rangle && (r_\mathbf{d} = q \overset{\mathbf{d}}{\lhd} p) \\
&= \sum_{m \in \mathsf{M}_p^{sq}(X)} \sum_{n \in \mathsf{M}_q^{sq}(p_m(X))} |q_n(p_m(X))\rangle && \text{(via Thm. 2)} \\
&\overset{(16)}{=} \sum_{m \in \mathbf{M}_p^{sq}(X)} \rho_{\mathbf{C}}^{sq}(\delta(q))|p_m(X)\rangle \\
&\overset{(16)}{=} \rho_{\mathbf{C}}^{sq}(\delta(q))\rho_{\mathbf{C}}^{sq}(\delta(p))|X\rangle\,.
\end{aligned}$$

$\square$

B.5 Proof of the SqPO stochastic mechanics framework theorem

Proof. By definition, the SqPO-type canonical representation of a generic rule algebra element $(O \overset{p}{\Leftarrow} I) \in \mathscr{R}_{\mathbf{C}}$ is a row-finite linear operator, since by virtue of the finitarity of objects according to Assumption 2, for every object $X \in \mathrm{obj}(\mathbf{C})$ the set of SqPO-admissible matches $\mathsf{M}_p^{sq}(X)$ of the associated linear rule $p = (O \overset{o}{\leftarrow} K \overset{i}{\rightarrow} I)$ is finite. We may thus verify that the linear operator H possesses all the required properties of a so-called Q-matrix (or infinitesimal generator) of a CTMC [41, 1], i.e. its non-diagonal entries are non-negative, its diagonal entries are finite, and furthermore the row sums of H are zero (whence H constitutes a conservative and stable Q-matrix; compare (21) of Definition 9). It is crucial to note that while originally H as a linear combination of representations of rule algebra elements is only defined to act on *finite* linear combinations of basis vectors $|X\rangle$ of $\hat{\mathbf{C}}$, an important mathematical result from the theory of CTMCs entails that if a row-finite linear operator such as H is a stable and conservative Q-matrix, it extends to a linear operator on *infinitely supported distributions* (here over basis vectors of

$\hat{\mathbf{C}}$) with finite real coefficients (see e.g. [1], Chapters 1 and 2). Moreover, the property $\langle| H = 0$ follows directly from the defining equations (21) of Definition 9.

Let us prove next the claim on the precise structure of observables. Recall that according to Definition 10, an observable $O \in \mathscr{O}_{\mathbf{C}}$ must be a linear operator in $End(\mathscr{S}_{\mathbf{C}})$ that acts diagonally on basis states $|X\rangle$ (for $X \in \mathrm{obj}(\mathbf{C})_{\cong}$), whence that satisfies for all $X \in \mathrm{obj}(\mathbf{C})_{\cong}$

$$O|X\rangle = \omega_O(X)|X\rangle \quad (\omega_O(X) \in \mathbb{R}).$$

Comparing this equation to the definition of the SqPO-type canonical representation (Definition 8) of a generic rule algebra basis element $\delta(p) \in \mathscr{R}_{\mathbf{C}}^{sq}$ (for $p \equiv (O \xleftarrow{o} K \xrightarrow{i} I) \in \mathrm{Lin}(\mathbf{C})$),

$$\rho_{\mathbf{C}}^{sq}(\delta(p))|X\rangle := \begin{cases} \sum_{m \in \mathsf{M}_p^{sq}(X)} |p_m(X)\rangle & \text{if } \mathsf{M}_p^{sq}(X) \neq \emptyset \\ 0_{\hat{\mathbf{C}}} & \text{else,} \end{cases}$$

we find that in order for $\rho_{\mathbf{C}}^{sq}(\delta(p))$ to be diagonal we must have

$$\forall X \in \mathrm{obj}(\mathbf{C}) : \forall m \in \mathsf{M}_p^{sq}(X) : \quad p_m(X) \cong X.$$

But by definition of SqPO-type derivations of objects along admissible matches (Definition 4), the only linear rules $p \in \mathrm{Lin}(\mathbf{C})$ that have this special property are precisely the rules of the form

$$p_{id_M} = (M \xleftarrow{id_M} M \xrightarrow{id_M} M).$$

In particular, defining $O_M^{sq} := \rho_{\mathbf{C}}^{sq}(\delta(p_{id_M}))$, we find that the eigenvalue $\omega_{O_M^{sq}}(X)$ coincides with the cardinality of the set $\mathsf{M}_{p_{id_M}}^{sq}(X)$ of SqPO-admissible matches,

$$\forall X \in ob(\mathbf{C}) : \quad O_M^{sq}|X\rangle = |\mathsf{M}_{p_{id_M}}^{sq}(X)| \cdot |X\rangle.$$

This proves that the operators O_M^{sq} form a basis of diagonal operators on $End(\hat{\mathbf{C}})$ (and thus on $End(\mathscr{S}_{\mathbf{C}})$) that can arise as linear combinations of representations of rule algebra elements.

To prove the jump-closure property, note that it follows from Definition 4 that for an arbitrary *linear* rule $p \equiv (O \xleftarrow{o} K \xrightarrow{i} I) \in \mathrm{Lin}(\mathbf{C})$, a generic object $X \in \mathrm{obj}(\mathbf{C})$ and a monomorphism $m : I \to X$, m is according to Definition 4 both a match of the rule p as well as of the rule p_{id_I}. Evidently, the application of the rule p to X along the match m produces an object $p_m(X)$ that is in general different from the object $p_{id_{I_m}}(X)$ produced by application of the rule p_{id_I} to X along the match m. But by definition of the projection operator $\langle|$ (Definition 10),

$$\forall X \in \mathrm{obj}(\mathbf{C})_{\cong} : \quad \langle|X\rangle := 1_{\mathbb{R}},$$

we find that

$$\langle| p_m(X)\rangle = \langle| p_{id_{I_m}}(X)\rangle = 1,$$

whence we may prove the claim of the SqPO-type jump-closure property via verifying it on arbitrary basis elements (with notations as above):

$$\langle|\rho_{\mathbf{C}}^{sq}(\delta(p))|X\rangle = |\mathsf{M}_p^{sq}(X)| = |\mathsf{M}_{p_{id_I}}^{sq}(X)| = \langle|\rho_{\mathbf{C}}^{sq}(\delta(p_{id_I}))|X\rangle.$$

Since $X \in \mathrm{obj}(\mathbf{C})_{\cong}$ was chosen arbitrarily, we thus have indeed that

$$\langle|\rho_{\mathbf{C}}^{sq}(\delta(p)) = \langle|\rho_{\mathbf{C}}^{sq}(\delta(p_{id_I})).$$

This concludes the proof that our definition of continuous-time Markov chains based upon SqPO-type rewriting rules is well-posed and yields all the requisite properties. $\square$