



# Rewriting Theory for the Life Sciences

Nicolas Behr, Jean Krivine

## ► To cite this version:

Nicolas Behr, Jean Krivine. Rewriting Theory for the Life Sciences: A Unifying Theory of CTMC Semantics. International Conference on Graph Transformation 2020, Jun 2020, Bergen, Norway. pp.185-202, 10.1007/978-3-030-51372-6\_11 . hal-02920023

**HAL Id: hal-02920023**

**<https://hal.science/hal-02920023>**

Submitted on 18 Dec 2020

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Rewriting Theory for the Life Sciences: A Unifying Theory of CTMC Semantics<sup>\*</sup>

Nicolas Behr<sup>1</sup>✉ and Jean Krivine<sup>2</sup>

<sup>1</sup> Center for Research and Interdisciplinarity (CRI)

Université de Paris, INSERM U1284  
8-10 Rue Charles V, 75004 Paris, France

[nicolas.behr@cri-paris.org](mailto:nicolas.behr@cri-paris.org)

<sup>2</sup> Institut de Recherche en Informatique Fondamentale (IRIF)

Université de Paris, CNRS UMR 8243  
8 Place Aurélie Nemours, 75205 Paris Cedex 13, France  
[jean.krivine@irif.fr](mailto:jean.krivine@irif.fr)

**Abstract.** The Kappa biochemistry and the MØD organo-chemistry frameworks are amongst the most intensely developed applications of rewriting theoretical methods in the life sciences to date. A typical feature of these types of rewriting theories is the necessity to implement certain structural constraints on the objects to be rewritten (a protein is empirically found to have a certain signature of sites, a carbon atom can form at most four bonds, ...). In this paper, we contribute to the theoretical foundations of these types of rewriting theory a number of conceptual and technical developments that permit to implement a universal theory of continuous-time Markov chains (CTMCs) for stochastic rewriting systems. Our core mathematical concepts are a novel rule algebra construction for the relevant setting of rewriting rules with conditions, both in Double- and in Sesqui-Pushout semantics, augmented by a suitable stochastic mechanics formalism extension that permits to derive dynamical evolution equations for pattern-counting statistics.

**Keywords:** Double-Pushout rewriting · Sesqui-pushout rewriting · rule algebra · stochastic mechanics · biochemistry · organic chemistry.

## 1 Motivation

One of the key applications that rewriting theory may be considered for in the life sciences is the theory of continuous-time Markov chains (CTMCs) modeling complex systems. In fact, since Delbrück’s seminal work on autocatalytic reaction systems in the 1940s [19], the mathematical theory of chemical reaction systems has effectively been formulated as a rewriting theory in disguise, namely via the rule algebra of discrete graph rewriting [10]. In the present paper, we provide the necessary technical constructions in order to consider

---

<sup>\*</sup> This is an extended version (containing additional technical appendices) of a paper with the same title accepted for [ICGT 2020](#).

the CTMCs and analysis methods of relevance for more general types of compositional rewriting theories with conditions, with key examples provided in the form of *biochemical graph rewriting* in the sense of the KAPPA framework (<https://kappalanguage.org>) [11], and (*organo-*) *chemical graph rewriting* in the sense of the MØD framework (<https://cheminf.imada.sdu.dk/mod/>) [1]. The present paper aims to serve two main purposes: the first consists in providing an extension of the existing category-theoretical rule-algebra frameworks [9,10,4] by the rewriting theoretical design feature of incorporating rules with conditions as well as constraints on objects (Section 3). Based upon these technical developments, we then investigate to which extent it is possible to utilize the rule-algebraic *stochastic mechanics frameworks* of the relevant types (Section 4) in order to compute evolution equations for the moments of pattern-count observables within the KAPPA and MØD frameworks (Section 5 and 6).

## 2 Compositional rewriting theories with conditions

The well-established *Double-Pushout* (DPO) [20] and *Sesqui-Pushout* (SqPO) [12] frameworks for rewriting systems over categories with suitable adhesivity properties [30,24,26,22] provide a principled and very general foundation for rewriting theories. In practice, many applications require the rewriting of objects that are not part of an adhesive category themselves, but which may be obtained from a suitable “ambient” category via the notion of *conditions* on objects. Together with a corresponding notion of constraints on rewriting rules, this yields a versatile extension of rewriting theory. In the DPO setting, this modification had been well-known [27,20,22,21], while it has been only very recently introduced for the SqPO setting [8]. For the *rule algebra* constructions presented in the main part of this contribution, we require in addition a certain *compositionality* property of our rewriting theories (established for the DPO case in [9,10], for the SqPO case in [4], and for both settings augmented with conditions in [8]).

### 2.1 Category-theoretical prerequisites

We collect in Appendix A.1 some of the salient concepts on  $\mathcal{M}$ -adhesive categories and the relevant notational conventions. Throughout this paper, we will make the following assumptions:

**Assumption 1.**  $\mathbf{C} \equiv (\mathbf{C}, \mathcal{M})$  is a finitary  $\mathcal{M}$ -adhesive category with  $\mathcal{M}$ -initial object,  $\mathcal{M}$ -effective unions and epi- $\mathcal{M}$ -factorization. In the setting of Sesqui-Pushout (SqPO) rewriting, we assume in addition that all final pullback complements (FPCs) along composable pairs of  $\mathcal{M}$ -morphisms exist, and that  $\mathcal{M}$ -morphisms are stable under FPCs.

Both of the main application examples presented within this paper rely upon typed variants of undirected multigraphs.

**Definition 1.** Let  $\mathcal{P}^{(1,2)} : \mathbf{Set} \rightarrow \mathbf{Set}$  be the restricted powerset functor (mapping a set  $S$  to the set of its subsets  $P \subset S$  with  $1 \leq |P| \leq 2$ ). The category

**uGraph** [10] of finite undirected multigraphs is defined as the finitary restriction of the comma category  $(ID_{\mathbf{Set}}, \mathcal{P}^{(1,2)})$ . Thus an undirected multigraph is specified as  $G = (E_G, V_G, i_G)$ , where  $E_G$  and  $V_G$  are (finite) sets of edges and vertices, respectively, and where  $i_G : E_G \rightarrow \mathcal{P}^{(1,2)}(V_G)$  is the edge-incidence map.

**Theorem 1.** **uGraph** satisfies Assumption 1, both for the DPO- and for the extended SqPO-variant.

*Proof.* As demonstrated in [10], **uGraph** is indeed a finitary  $\mathcal{M}$ -adhesive category with  $\mathcal{M}$ -initial object and  $\mathcal{M}$ -effective unions, for  $\mathcal{M}$  the class of component-wise monic **uGraph**-morphisms. It thus remains to prove the existence of an epi- $\mathcal{M}$ -factorization as well as the properties related to FPCs. To this end, utilizing the fact that the category **Set** upon which the comma category **uGraph** is based possesses an epi-mono-factorization, we may construct the following diagram from a **uGraph**-morphism  $\varphi = (\varphi_E, \mathcal{P}^{(1,2)}(\varphi_V))$  (for component morphisms  $\varphi_E : E \rightarrow E'$  and  $\varphi_V : V \rightarrow V'$ ):

$$\begin{array}{ccccc}
 E & \xrightarrow{i} & \mathcal{P}^{(1,2)}(V) & \xleftarrow{\mathcal{P}^{(1,2)}} & V \\
 \downarrow e_E & \searrow e_P & \downarrow \mathcal{P}^{(1,2)}(e_V) & & \downarrow e_V \\
 \varphi_E \downarrow \overline{E} & \xrightarrow{\exists \cong} & \overline{\overline{E}} & \xrightarrow{m_P} & P \\
 \downarrow m_E & \nearrow p_E & & \downarrow p_V & \downarrow \mathcal{P}^{(1,2)}(m_V) \\
 E' & \xrightarrow{i'} & \mathcal{P}^{(1,2)}(V') & \xleftarrow{\mathcal{P}^{(1,2)}} & V' \\
 & & \downarrow \mathcal{P}^{(1,2)}(m_V) & & \downarrow m_V \\
 & & \mathcal{P}^{(1,2)}(\overline{V}) & \xleftarrow{\mathcal{P}^{(1,2)}} & \overline{V} \\
 & & \downarrow m_V & & \downarrow m_V \\
 & & V' & & V
 \end{array} \quad (1)$$

The diagram is constructed as follows:

1. Perform the epi-mono-factorizations  $\varphi_E = m_E \circ e_E$  and  $\varphi_V = m_V \circ e_V$ , and apply the functor  $\mathcal{P}^{(1,2)}$  in order to obtain the morphisms  $\mathcal{P}^{(1,2)}(e_V)$  and  $\mathcal{P}^{(1,2)}(m_V)$ ; since the functor  $\mathcal{P}^{(1,2)}$  preserves monomorphisms [33],  $\mathcal{P}^{(1,2)}(m_V) \in \mathbf{mono}(\mathbf{Set})$ .
2. Construct the pullback

$$(E' \leftarrow P \rightarrow \mathcal{P}^{(1,2)}(\overline{V})) := \mathbf{PB}(E' \rightarrow \mathcal{P}^{(1,2)}(V') \leftarrow \mathcal{P}^{(1,2)}(\overline{V})),$$

Since monomorphisms are stable under pullback in **Set**, having proved that  $\mathcal{P}^{(1,2)}(m_V) \in \mathbf{mono}(\mathbf{Set})$  implies  $(p_E : P \rightarrow E') \in \mathbf{mono}(\mathbf{Set})$ .

3. By the universal property of pullbacks, there exists a morphism  $(p : E \rightarrow P)$ . Let  $p = m_P \circ e_P$  be the epi-mono-factorization of this morphism.
4. By stability of monomorphisms under composition in **Set**, we find that  $p_E \circ m_P \in \mathbf{mono}(\mathbf{Set})$ , and consequently  $\varphi_E = (p_E \circ m_P) \circ e_P$  yields an alternative epi-mono-factorization of  $\varphi_E$ . Then by uniqueness of epi-mono-factorizations up to isomorphism, there must exist an isomorphism  $(\overline{E} \rightarrow \overline{\overline{E}}) \in \mathbf{iso}(\mathbf{Set})$ .

We have thus demonstrated that both  $(e_E, \mathcal{P}^{(1,2)}(e_V))$  and  $(m_E, \mathcal{P}^{(1,2)}(m_V))$  are morphisms in **uGraph**. Since morphisms in comma categories are mono-, epi- or iso-morphisms if they are so componentwise [20], we conclude that

$$(e_E, \mathcal{P}^{(1,2)}(e_V)) \in \text{epi}(\mathbf{uGraph}), \quad (m_E, \mathcal{P}^{(1,2)}(m_V)) \in \text{mono}(\mathbf{uGraph}),$$

which finally entails that we have explicitly constructed an epi-mono-factorization of the **uGraph**-morphism  $(\varphi_E, \mathcal{P}^{(1,2)}(\varphi_V))$ .

In order to demonstrate that FPCs along pairs of composable  $\mathcal{M}$ -morphisms  $\varphi_A, \varphi_B \in \mathcal{M}$  in **uGraph** exist (for  $\mathcal{M}$  the class of component-wise monomorphic **uGraph** morphisms), we provide the following explicit construction:

$$\begin{array}{ccc|l} A & \xrightarrow{\varphi_A} & B & \\ \varphi_C \downarrow & \text{FPC} & \downarrow \varphi_B & \\ C & \xrightarrow{\varphi_D} & D & \end{array} \quad \left| \begin{array}{l} V_C = V_D \setminus (V_B \setminus V_A) \\ E_C = \{e \in E_D \setminus (E_B \setminus E_A) \mid u_D(e) \in \mathcal{P}^{(1,2)}(V_C)\} \\ u_C = u_D|_{E_C} \\ \varphi_C = (E_A \hookrightarrow E_C, \mathcal{P}^{(1,2)}(V_A \hookrightarrow V_C)) \\ \varphi_D = (E_C \hookrightarrow E_D, \mathcal{P}^{(1,2)}(V_C \hookrightarrow V_D)) \end{array} \right. \quad (2)$$

## 2.2 Conditions

Referring to Appendix A.2 for further details and technical definitions, we will utilize as a **notational convention** the standard shorthand notations

$$\exists(X \hookrightarrow Y) := \exists(X \hookrightarrow Y, \text{true}_Y), \quad \forall(X \hookrightarrow Y, c_Y) := \neg \exists(X \hookrightarrow Y, \neg c_Y). \quad (3)$$

For example, the constraints

$$c_{\emptyset}^{(1)} = \exists(\emptyset \hookrightarrow \bullet \quad \bullet), \quad c_{\emptyset}^{(2)} = \neg \exists(\emptyset \hookrightarrow \bullet \leftrightarrow \bullet), \quad c_{\emptyset}^{(3)} = \forall(\emptyset \hookrightarrow \bullet \leftarrow \bullet, \exists(\bullet \leftarrow \bullet \hookrightarrow \bullet \leftrightarrow \bullet))$$

express for a given object  $Z \in \text{obj}(\mathbf{C})$  that  $Z$  contains at least two vertices (if  $Z \models c_{\emptyset}^{(1)}$ ), that  $Z$  does not contain parallel pairs of directed edges (if  $Z \models c_{\emptyset}^{(2)}$ ), and that for every directed edge in  $Z$  there also exists a directed edge between the same endpoints with opposite direction (if  $Z \models c_{\emptyset}^{(3)}$ ), respectively.

## 2.3 Compositional rewriting with conditions

Throughout this section, we assume that we are given a type  $\mathbb{T} \in \{DPO, SqPO\}$  of rewriting semantics and an  $\mathcal{M}$ -adhesive category  $\mathbf{C}$  satisfying the respective variant of Assumption 1. In categorical rewriting theories, the universal constructions utilized such as pushouts, pullbacks, pushout complements and final pullback complements are unique only up to universal isomorphisms. This motivates specifying a suitable notion of equivalence classes of rules with conditions:

**Definition 2 (Rules with conditions).** Let  $\overline{\text{Lin}}(\mathbf{C})$  denote the class of (linear) rules with conditions, defined as

$$\overline{\text{Lin}}(\mathbf{C}) := \{(O \xleftarrow{o} K \xrightarrow{i} I; c_I) \mid o, i \in \mathcal{M}, c_I \in \text{cond}(\mathbf{C})\}. \quad (4)$$

We define two rules with conditions  $R_j = (r_j, c_{I_j})$  ( $j = 1, 2$ ) equivalent, denoted  $R_2 \sim R_1$ , iff  $c_{I_1} \equiv c_{I_2}$  and if there exist isomorphisms  $\omega, \kappa, \iota \in \text{iso}(\mathbf{C})$  such that the diagram on the right commutes. We denote by  $\overline{\text{Lin}}(\mathbf{C})_\sim$  the set of equivalence classes under  $\sim$  of rules with conditions.

$$\begin{array}{ccccc} O_1 & \longleftarrow & K_1 & \longrightarrow & I_1 \\ \cong \downarrow \omega & & \cong \downarrow \kappa & & \cong \downarrow \iota \\ O_2 & \longleftarrow & K_2 & \longrightarrow & I_2 \end{array} \quad (5)$$

**Definition 3 (Direct derivations).** Let  $r = (O \leftrightarrow K \hookrightarrow I) \in \text{Lin}(\mathbf{C})$  and  $c_I \in \text{cond}(\mathbf{C})$  be concrete representatives of some equivalence class  $R \in \overline{\text{Lin}}(\mathbf{C})_\sim$ , and let  $X, Y \in \text{obj}(\mathbf{C})$  be objects. Then a type  $\mathbb{T}$  direct derivation is defined as a commutative diagram such as below right, where all morphism are in  $\mathcal{M}$  (and with the left representation a shorthand notation)

$$\begin{array}{ccc} O & \xleftarrow{r} & I \\ m^* \downarrow & \mathbb{T} & \downarrow m \\ Y & \longleftarrow & X \end{array} \quad := \quad \begin{array}{ccccc} O & \longleftarrow & K & \longrightarrow & I \\ m^* \downarrow & (B) & \downarrow k & (A) & \downarrow m \\ Y & \longleftarrow & \overline{K} & \longrightarrow & X \end{array} \quad (6)$$

with the following pieces of information required relative to the type:

1.  $\mathbb{T} = \mathbf{DPO}$ : given  $(m : I \hookrightarrow X) \in \mathcal{M}$ ,  $m$  is a DPO-admissible match of  $R$  into  $X$ , denoted  $m \in \mathbf{M}_R^{\text{DPO}}(X)$ , if  $m \models c_I$  and  $(A)$  is constructable as a pushout complement, in which case  $(B)$  is constructed as a pushout.
2.  $\mathbb{T} = \mathbf{SqPO}$ : given  $(m : I \hookrightarrow X) \in \mathcal{M}$ ,  $m$  is a SqPO-admissible match of  $R$  into  $X$ , denoted  $m \in \mathbf{M}_R^{\text{SqPO}}(X)$ , if  $m \models c_I$ , in which case  $(A)$  is constructed as a final pullback complement and  $(B)$  as a pushout.
3.  $\mathbb{T} = \mathbf{DPO}^\dagger$ : given just the “plain rule”  $r$  and  $(m^* : O \hookrightarrow Y) \in \mathcal{M}$ ,  $m^*$  is a DPO $^\dagger$ -admissible match of  $r$  into  $X$ , denoted  $m \in \mathbf{M}_r^{\text{DPO}^\dagger}(Y)$ , if  $(B)$  is constructable as a pushout complement, in which case  $(B)$  is constructed as a pushout.

For types  $\mathbb{T} \in \{\text{DPO}, \text{SqPO}\}$ , we will sometimes employ the notation  $R_m(X)$  for the object  $Y$ .

Note that at this point, we have not yet resolved a conceptual issue that arises from the non-uniqueness of a direct derivation given a rule and an admissible match. Concretely, the pushout complement, pushout and FPC constructions are only unique up to isomorphisms. This issue will ultimately be resolved as part of the rule algebraic theory. We next consider a certain *composition operation* on rules with conditions that is quintessential to our main constructions:

**Definition 4 (Rule compositions).** Let  $R_1, R_2 \in \overline{\text{Lin}}(\mathbf{C})_\sim$  be two equivalence classes of rules with conditions, and let  $r_j \in \text{Lin}(\mathbf{C})$  and  $c_{I_j}$  be concrete representatives of  $R_j$  (for  $j = 1, 2$ ). For  $\mathbb{T} \in \{\text{DPO}, \text{SqPO}\}$ , an  $\mathcal{M}$ -span  $\mu = (I_2 \hookleftarrow M_{21} \hookrightarrow O_1)$  (i.e. with  $(M_{21} \hookrightarrow O_1), (M_{21} \hookrightarrow I_2) \in \mathcal{M}$ ) is a  $\mathbb{T}$ -admissible match of  $R_2$  into  $R_1$  if the diagram below is constructable (with  $N_{21}$

constructed by taking pushout)

$$\begin{array}{ccccccc}
 O_2 & \xleftarrow{r_2} & I_2 & \xleftrightarrow{\quad} & M_{21} & \xrightarrow{\quad} & O_1 \xleftarrow{r_1} I_1 \\
 \downarrow & & \mathbb{T} & & \text{PO} & & \text{DPO}^\dagger \downarrow \\
 O_{21} & \xleftarrow{\quad} & N_{21} & \xleftrightarrow{\quad} & & \xleftarrow{\quad} & I_{21}
 \end{array} \quad (7)$$

and if  $c_{I_{21}} \neq \text{false}$ . Here, the condition  $c_{I_{21}}$  is computed as

$$c_{I_{21}} := \text{Shift}(I_1 \hookrightarrow I_{21}, c_{I_1}) \wedge \text{Trans}(N_{21} \leftarrow I_{21}, \text{Shift}(I_2 \hookrightarrow N_{21}, c_{I_2})). \quad (8)$$

In this case, we define the type  $\mathbb{T}$  composition of  $R_2$  with  $R_1$  along  $\mu$ , denoted  $R_2^{\mu \triangleleft_{\mathbb{T}}} R_1$ , as

$$R_2^{\mu \triangleleft_{\mathbb{T}}} R_1 := [(O_{21} \leftarrow I_{21}; c_{I_{21}})]_{\sim}, \quad (9)$$

where  $(O_{21} \leftarrow I_{21}) := (O_{21} \leftarrow N_{21}) \circ (N_{21} \leftarrow I_{21})$  (with  $\circ$  the span composition operation).

We recall in Appendix A.3 two important technical results on the notions of direct derivations and rule compositions that have been derived in [8] (where however the DPO-type concurrency theorem is of course classical, cf. e.g. [20]).

### 3 Rule algebras for compositional rewriting with conditions

The associativity property of rule compositions in both DPO- and SqPO-type semantics for rewriting with conditions as proved in [8] may be fruitfully exploited within rule algebra theory. One possibility to encode the non-determinism in sequential applications of rules to objects is given by lifting each possible configuration  $X \in \text{obj}(\mathbf{C})_{\cong}$  (i.e. isomorphism class of *objects*) to a basis vector  $|X\rangle$  of a vector space  $\hat{\mathbf{C}}$ ; then a rule  $r$  is lifted to a linear operator acting on  $\hat{\mathbf{C}}$ , with the idea that this operator acting upon a basis vector  $|X\rangle$  should evaluate to the “sum over all possibilities to act with  $r$  on  $X$ ”. We will extend here the general rule algebra framework [6,9,4] to the present setting of rewriting rules with conditions.

We will first lift the notion of rule composition into the setting of a composition operation on a certain abstract vector space over rules, thus realizing the heuristic concept of “summing over all possibilities to compose rules”.

**Definition 5.** Let  $\mathbb{T} \in \{\text{DPO}, \text{SqPO}\}$  be the rewriting type, and let  $\mathbf{C}$  be a category satisfying the relevant variant of Assumption 1. Let  $\overline{\mathcal{R}}_{\mathbf{C}}$  be an  $\mathbb{R}$ -vector space, defined via a bijection  $\delta : \overline{\text{Lin}}(\mathbf{C})_{\sim} \xrightarrow{\cong} \text{basis}(\overline{\mathcal{R}}_{\mathbf{C}})$  from the set of equivalence classes of linear rules with conditions to the set of basis vectors of  $\overline{\mathcal{R}}_{\mathbf{C}}$ . Let  $\star_{\mathbb{T}}$  denote the type  $\mathbb{T}$  rule algebra product, a binary operation defined via its action on basis elements  $\delta(R_1), \delta(R_2) \in \overline{\mathcal{R}}_{\mathbf{C}}$  (for  $R_1, R_2 \in \overline{\text{Lin}}(\mathbf{C})_{\sim}$ ) as

$$\delta(R_2) \star_{\mathbb{T}} \delta(R_1) := \sum_{\mu \in \mathcal{M}_{R_2}^{\mathbb{T}}(R_1)} \delta(R_2^{\mu \triangleleft_{\mathbb{T}}} R_1). \quad (10)$$

We refer to  $\overline{\mathcal{R}}_{\mathbf{C}}^{\mathbb{T}} := (\overline{\mathcal{R}}_{\mathbf{C}}, \star_{\mathbb{T}})$  as the  $\mathbb{T}$ -type rule algebra over  $\mathbf{C}$ .

**Theorem 2.** For type  $\mathbb{T} \in \{DPO, SqPO\}$  over a category  $\mathbf{C}$  satisfying Assumption 1, the rule algebra  $\overline{\mathcal{R}}_{\mathbf{C}}^{\mathbb{T}}$  is an associative unital algebra, with unit element  $\delta(R_{\emptyset})$ , where  $R_{\emptyset} := (\emptyset \leftarrow \emptyset \hookrightarrow \emptyset; \text{true})$ .

*Proof.* Associativity follows from Theorem 7, while unitality, i.e. that

$$\forall R \in \overline{\text{Lin}}(\mathbf{C})_{\sim} : \quad \delta(R_{\emptyset}) \star_{\mathbb{T}} \delta(R) = \delta(R) \star_{\mathbb{T}} \delta(R_{\emptyset}) = \delta(R)$$

follows directly from an explicit computation of the relevant rule compositions.

As alluded to in the introduction, the prototypical example of rule algebras are those of DPO- or (in this case equivalently) SqPO-type over discrete graphs, giving rise as a special case to the famous Heisenberg-Weyl algebra of key importance in mathematical chemistry, combinatorics and quantum physics (see [10] for further details). We will now illustrate the rule algebra concept in an example involving a more general base category.

*Example 1.* For the category **uGraph** and DPO-type rewriting semantics, consider as an example the following two rules with conditions:

$$R_C := \left( \begin{array}{c} \bullet \\ \vdots \end{array} \leftarrow \begin{array}{c} \bullet \\ \vdots \end{array} \hookrightarrow \begin{array}{c} \bullet \\ \vdots \end{array} ; \neg \exists \left( \begin{array}{c} \bullet \\ \vdots \end{array} \hookrightarrow \begin{array}{c} \bullet \\ \vdots \end{array} \right) \right), \quad R_V := (\bullet \leftarrow \emptyset \hookrightarrow \bullet; \text{true}). \quad (11)$$

The first rule is a typical example of a rule with application conditions, i.e. here stating that the rule may only link two vertices if they were previously not already linked to each other. The second rule, owing to DPO semantics, can in effect only be applied to vertices without any incident edges. The utility of the rule-algebraic composition operation then consists in reasoning about sequential compositions of these rules, for example (letting  $*$  :=  $\star_{DPO}$ ):

$$\begin{aligned} \delta(R_C) * \delta(R_V) &= \delta(R_C \uplus R_V) + 2\delta(R'_C), \quad R'_C := \left( \begin{array}{c} \bullet \\ \vdots \end{array} \leftarrow \begin{array}{c} \bullet \\ \vdots \end{array} \hookrightarrow \begin{array}{c} \bullet \\ \vdots \end{array} ; \text{true} \right) \\ \delta(R_V) * \delta(R_C) &= \delta(R_C \uplus R_V). \end{aligned} \quad (12)$$

To provide some intuition: the first computation encodes the causal information that the two rules may either be composed along a trivial overlap, or rule  $R_C$  may overlap on one of the vertices in the output of  $R_V$ ; in the latter case, any vertex to which first  $R_V$  and then  $R_C$  applies must not have had any incident edges, i.e. in particular no edge violating the constraint of  $R_C$ , which is why the composite rule  $R'_C$  does not feature any non-trivial constraint. In the other order of composition, the two vertices in the output of  $R_C$  are linked by an edge, so  $R_V$  cannot be applied to any of these two vertices (leaving just the trivial overlap contribution).

Just as the rule algebra construction encodes the compositional associativity property of rule compositions, the following *representation* construction encodes in a certain sense the properties described by the concurrency theorem:



**Definition 6.** Let  $\mathbf{C}$  be an  $\mathcal{M}$ -adhesive category satisfying Assumption 1. Let  $\hat{\mathbf{C}}$  be defined as the  $\mathbb{R}$ -vector space whose set of basis vectors is isomorphic to the set<sup>3</sup> of iso-classes of objects of  $\mathbf{C}$  via a bijection  $|\cdot\rangle : \text{obj}(\mathbf{C})_{\cong} \rightarrow \text{basis}(\hat{\mathbf{C}})$ . Then the  $\mathbb{T}$ -type canonical representation of the  $\mathbb{T}$ -type rule algebra over  $\mathbf{C}$ , denoted  $\bar{\rho}_{\mathbf{C}}^{\mathbb{T}}$ , is defined as the morphism  $\bar{\rho}_{\mathbf{C}}^{\mathbb{T}} : \bar{\mathcal{R}}_{\mathbf{C}}^{\mathbb{T}} \rightarrow \text{End}_{\mathbb{R}}(\hat{\mathbf{C}})$  specified via

$$\forall R \in \overline{\text{Lin}}(\mathbf{C})_{\sim}, X \in \text{obj}(\mathbf{C})_{\cong} : \quad \bar{\rho}_{\mathbf{C}}^{\mathbb{T}}(\delta(R)) |X\rangle := \sum_{m \in \mathcal{M}_R^{\mathbb{T}}(X)} |R_m(X)\rangle. \quad (13)$$

**Theorem 3.**  $\bar{\rho}_{\mathbf{C}}^{\mathbb{T}}$  as defined above is an algebra homomorphism (and thus in particular a well defined representation).

*Proof.* The proof is entirely analogous to the one for the case without application conditions [9,4] (cf. Appendix B.1).

## 4 Stochastic mechanics formalism

Referring to [6,10,7] for further details and derivations, suffice it here to highlight the key role played by the algebraic concept of *commutators* in stochastic mechanics. Let us first provide the constructions of continuous-time Markov chains (CTMCs) and observables in stochastic rewriting systems.

**Definition 7.** Let  $\langle | : \hat{\mathbf{C}} \rightarrow \mathbb{R}$  (referred to as dual projection vector) be defined via its action on basis vectors of  $\hat{\mathbf{C}}$  as  $\langle |X\rangle := 1_{\mathbb{R}}$ .

**Theorem 4.** Let  $\mathbf{C}$  be a category satisfying the relevant variant of Assumption 1, and let  $\bar{\mathcal{R}}_{\mathbf{C}}^{\mathbb{T}}$  be the  $\mathbb{T}$ -type rule algebra of linear rules with conditions over  $\mathbf{C}$ . Let  $\rho \equiv \bar{\rho}_{\mathbf{C}}^{\mathbb{T}}$  denote the  $\mathbb{T}$ -type canonical representation of  $\bar{\mathcal{R}}_{\mathbf{C}}^{\mathbb{T}}$ . Then the following results hold:

1. The basis elements of the space  $\text{obs}(\mathbf{C})_{\mathbb{T}}$  of  **$\mathbb{T}$ -type observables**, i.e. the diagonal linear operators that arise as (linear combinations of)  $\mathbb{T}$ -type canonical representations of rewriting rules with conditions, have the following structure ( $\hat{\mathcal{O}}_{P,q}^{\mathcal{C}_P}$  in the DPO case,  $\hat{\mathcal{O}}_P^{\mathcal{C}_P}$  in the SqPO case):

$$\begin{aligned} \hat{\mathcal{O}}_{P,q}^{\mathcal{C}_P} &:= \rho(\delta(P \xleftarrow{q} Q \xrightarrow{q} P; \mathcal{C}_P)) \quad (P \in \text{obj}(\mathbf{C})_{\cong}, q \in \mathcal{M}, \mathcal{C}_P \in \text{cond}(\mathbf{C})_{\sim}) \\ \hat{\mathcal{O}}_P^{\mathcal{C}_P} &:= \rho(\delta(P \xleftarrow{\cong} P \xrightarrow{\cong} P; \mathcal{C}_P)) \quad (P \in \text{obj}(\mathbf{C})_{\cong}, \mathcal{C}_P \in \text{cond}(\mathbf{C})_{\sim}). \end{aligned} \quad (14)$$

2. **DPO-type jump closure property:** for every linear rule with condition  $R \equiv (O \leftarrow K \hookrightarrow I, \mathcal{C}_I) \in \overline{\text{Lin}}(\mathbf{C})$ , we find that

$$\langle | \rho(\delta(R)) = \langle | \hat{\mathcal{O}}(\delta(R)), \quad (15)$$

<sup>3</sup> We assume here that the isomorphism classes of objects of  $\mathbf{C}$  form a set (i.e. not a proper class).

where  $\hat{\mathbb{O}} : \overline{\mathcal{R}}_{\mathbf{C}}^{DPO} \rightarrow \text{End}_{\mathbb{R}}(\hat{\mathbf{C}})$  is the homomorphism defined via its action on basis elements  $\delta(R)$  for  $R = (O \leftarrow K \hookrightarrow I; \mathbf{c}_I) \in \overline{\text{Lin}}(\mathbf{C})_{\sim}$  as

$$\hat{\mathbb{O}}(\delta(R)) := \rho(\delta(I \leftarrow K \hookrightarrow I; \mathbf{c}_I)) \in \text{obs}(\mathbf{C}). \quad (16)$$

3. **SqPO-type jump closure property:** for every linear rule with condition  $R \equiv (O \leftarrow K \hookrightarrow I; \mathbf{c}_I) \in \overline{\text{Lin}}(\mathbf{C})$ , we find that

$$\langle | \rho(\delta(R)) = \langle | \hat{\mathbb{O}}(\delta(R)), \quad (17)$$

where<sup>4</sup>  $\hat{\mathbb{O}} : \overline{\mathcal{R}}_{\mathbf{C}}^{SqPO} \rightarrow \text{End}_{\mathbb{R}}(\hat{\mathbf{C}})$  is the homomorphism defined via

$$\hat{\mathbb{O}}(\delta(R)) := \rho(\delta(I \xleftarrow{\cong} I \xrightarrow{\cong} I; \mathbf{c}_I)) \in \text{obs}(\mathbf{C}). \quad (18)$$

4. **CTMCs via stochastic rewriting systems:** Let  $\text{Prob}(\mathbf{C})$  be the space of (sub-)probability distributions over  $\hat{\mathbf{C}}$  (i.e.  $|\Psi\rangle = \sum_{X \in \text{obj}(\mathbf{C})_{\cong}} \psi_X |X\rangle$ ). Let  $\mathcal{T}$  be a collection of  $N$  pairs of positive real-valued parameters  $\kappa_j$  (referred to as base rates) and linear rules  $R_j$  with application conditions,

$$\mathcal{T} := \{(\kappa_j, R_j)\}_{1 \leq j \leq N} \quad (\kappa_j \in \mathbb{R}_{\geq 0}, R_j \equiv (r_j, \mathbf{c}_{I_j}) \in \overline{\text{Lin}}(\mathbf{C})). \quad (19)$$

Then given an initial state  $|\Psi_0\rangle \in \text{Prob}(\mathbf{C})$ , the  $\mathbb{T}$ -type stochastic rewriting system based upon the transitions  $\mathcal{T}$  gives rise to the CTMC  $(\mathcal{H}, |\Psi(0)\rangle)$  with time-dependent state  $|\Psi(t)\rangle \in \text{Prob}(\mathbf{C})$  (for  $t \geq 0$ ) and evolution equation

$$\forall t \geq 0 : \quad \frac{d}{dt} |\Psi(t)\rangle = \mathcal{H} |\Psi(t)\rangle, \quad |\Psi(0)\rangle = |\Psi_0\rangle. \quad (20)$$

Here, the infinitesimal generator  $\mathcal{H}$  of the CTMC is given by

$$\mathcal{H} = \hat{H} - \hat{\mathbb{O}}(\hat{H}), \quad \hat{H} = \sum_{j=1}^N \kappa_j \rho(\delta(R_j)). \quad (21)$$

*Proof.* See Appendix B.2.

*Remark 1.* The operation  $\hat{\mathbb{O}}$  featuring in the DPO- and SqPO-type jump-closure properties has a very intuitive interpretation: given a linear rule with condition  $R \equiv (r, \mathbf{c}_I) \in \overline{\text{Lin}}(\mathbf{C})$ , the linear operator  $\hat{\mathbb{O}}(\delta(R))$  is an observable that evaluates on a basis vector  $|X\rangle \in \hat{\mathbf{C}}$  as  $\hat{\mathbb{O}}(\delta(R)) |X\rangle = (\# \text{ of ways to apply } R \text{ to } X) \cdot |X\rangle$ .

As for the concrete computational techniques offered by the stochastic mechanics formalism, one of the key advantages of this rule-algebraic framework is the possibility to reason about *expectation values* (and higher moments) of pattern-count observables in a principled and universal manner. The precise formulation is given by the following generalization of results from [7] to the setting of DPO- and SqPO-type rewriting for rules with conditions:

<sup>4</sup> Since in applications we will always fix the type of rewriting to either DPO or SqPO, we will use the same symbol for the jump-closure operator in both cases.

**Theorem 5.** *Given a CTMC  $(|\Psi_0\rangle, \mathcal{H})$  with time-dependent state  $|\Psi(t)\rangle$  (for  $t \geq 0$ ), a set of observables  $O_1, \dots, O_n \in \text{obs}(\mathbf{C})$  and  $n$  formal variables  $\lambda_1, \dots, \lambda_n$ , define the exponential moment-generating function (EMGF)  $M(t; \underline{\lambda})$  as*

$$M(t; \underline{\lambda}) := \langle | e^{\underline{\lambda} \cdot \underline{O}} | \Psi(t) \rangle, \quad \underline{\lambda} \cdot \underline{O} := \sum_{j=1}^n \lambda_j O_j. \quad (22)$$

Then  $M(t; \underline{\lambda})$  satisfies the following formal evolution equation (for  $t \geq 0$ ):

$$\frac{d}{dt} M(t; \underline{\lambda}) = \sum_{q \geq 1} \frac{1}{q!} \langle | \left( ad_{\underline{\lambda} \cdot \underline{O}}^{oq}(\hat{H}) \right) e^{\underline{\lambda} \cdot \underline{O}} | \Psi(t) \rangle, \quad M(0; \underline{\lambda}) = \langle | e^{\underline{\lambda} \cdot \underline{O}} | \Psi_0 \rangle. \quad (23)$$

*Proof.* In full analogy to the case of rules without conditions [7], the proof follows from the BCH formula  $e^{\lambda A} B e^{-\lambda A} = e^{ad_{\lambda A}}(B)$  (for  $A, B \in \text{End}_{\mathbb{R}}(\hat{\mathbf{C}})$ ). Here,  $ad_A^0(B) := B$ ,  $ad_A(B) := AB - BA$  (also referred to as the *commutator*  $[A, B]$  of  $A$  and  $B$ ), and  $ad_A^{o(q+1)}(B) := ad_A(ad_A^{oq}(B))$  for  $q \geq 1$ . Finally, the  $q = 0$  term in the above expression evaluates identically to 0 due to  $\langle | \mathcal{H} = 0$ .

Combining this theorem with the notion of  $\mathbb{T}$ -type jump-closure, one can in favorable cases express the EMGF evolution equation as a PDE on formal power series in  $\lambda_1, \dots, \lambda_n$  and with  $t$ -dependent real-valued coefficients. Referring the interested readers to [7] for further details on this technique, let us provide here a simple non-trivial example of such a calculation.

*Example 2.* Let us consider a stochastic rewriting system over the category  $\mathbf{C} = \mathbf{uGraph}$  of finite undirected multigraphs, with objects further constrained by the structure constraint  $c_{\emptyset}^S := \neg \exists (\emptyset \hookrightarrow \bullet \circ \bullet) \in \text{cond}(\mathbf{uGraph})$  that prohibits multiedges. Let us consider for type  $\mathbb{T} = SqPO$  the four rules with conditions  $R_{E_{\pm}}$  (edge-creation/-deletion) and  $R_{V_{\pm}}$  (vertex creation/deletion), defined as

$$\begin{aligned} R_{E_+} &:= \frac{1}{2} \delta \left( \begin{array}{c} \bullet \\ | \\ \bullet \end{array} \leftarrow \begin{array}{c} \bullet \\ | \\ \bullet \end{array} \hookrightarrow \begin{array}{c} \bullet \\ | \\ \bullet \end{array} ; \neg \exists \left( \begin{array}{c} \bullet \\ | \\ \bullet \end{array} \hookrightarrow \begin{array}{c} \bullet \\ | \\ \bullet \end{array} \right) \right), \quad R_{V_+} := \delta(\bullet \leftarrow \emptyset \hookrightarrow \emptyset; \text{true}) \\ R_{E_-} &:= \frac{1}{2} \delta \left( \begin{array}{c} \bullet \\ | \\ \bullet \end{array} \leftarrow \begin{array}{c} \bullet \\ | \\ \bullet \end{array} \hookrightarrow \begin{array}{c} \bullet \\ | \\ \bullet \end{array} ; \text{true} \right), \quad R_{V_-} := \delta(\emptyset \leftarrow \emptyset \hookrightarrow \bullet; \text{true}). \end{aligned}$$

Here, the prefactors  $\frac{1}{2}$  for  $R_{E_{\pm}}$  are chosen purely for convenience. Note that  $R_{E_+}$  is the only rule requiring a non-trivial application condition, since linking two vertices with an edge might create a multiedge (precisely when the two vertices were already linked). Introducing base rates  $\nu_{\pm}, \varepsilon_{\pm} \in \mathbb{R}_{>0}$  and letting  $\hat{X} := \rho(R_X)$ , we may assemble the infinitesimal generator  $\mathcal{H}$  of a CTMC as

$$\mathcal{H} = \hat{H} + \hat{\mathbb{O}}(\hat{H}), \quad \hat{H} := \nu_+ \hat{V}_+ + \nu_- \hat{V}_- + \varepsilon_+ \hat{E}_+ + \varepsilon_- \hat{E}_-. \quad (24)$$

One might now ask whether there is any interesting dynamical structure e.g. in the evolution of the moments of the observables that count the number of times each of the transitions of this system is applicable,

$$O_{\bullet, \bullet} := \hat{\mathbb{O}}(\delta(R_C)), \quad O_{\bullet, -} := \hat{\mathbb{O}}(\delta(R_D)), \quad O_{\bullet} := \hat{\mathbb{O}}(\delta(R_{VD})). \quad (25)$$

The algebraic data necessary in order to formulate EMGF evolution equations are all **commutators** of the observables with the contributions  $\hat{X} := \rho(\delta(R_X))$  to the “off-diagonal part”  $\hat{H}$  of the infinitesimal generator  $\mathcal{H}$ . We will present here for brevity just those commutators necessary in order to compute the evolution equations for the averages of the three observables:

$$\begin{aligned} [O_\bullet, \hat{V}_\pm] &= \pm \hat{V}_\pm, & [O_\bullet, \hat{E}_\pm] &= 0 \\ [O_{\bullet|\bullet}, \hat{V}_+] &= \hat{A}, & [O_{\bullet|\bullet}, \hat{V}_-] &= -\hat{B}, & [O_{\bullet|\bullet}, \hat{E}_\pm] &= \mp \hat{E}_\pm \\ [O_{\bullet\bullet}, \hat{V}_+] &= 0, & [O_{\bullet\bullet}, \hat{V}_-] &= -\hat{C}, & [O_{\bullet\bullet}, \hat{E}_\pm] &= \pm \hat{E}_\pm \end{aligned} \quad (26)$$

As typical in these types of commutator computations, we find a number of contributions (here  $\hat{A}$ ,  $\hat{B}$  and  $\hat{C}$ ) that were not either observables or based upon rules of the SRS:

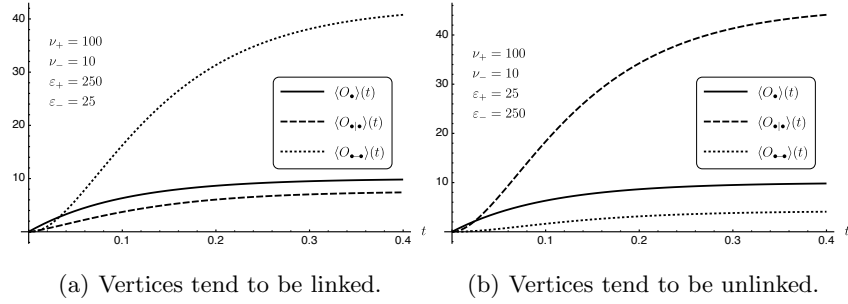
$$\begin{aligned} \hat{A} &:= \rho \left( \delta \left( \begin{array}{c} \bullet \\ \bullet \end{array} \leftarrow \begin{array}{c} \bullet \\ \bullet \end{array} \hookrightarrow \begin{array}{c} \bullet \\ \bullet \end{array}; \text{true} \right) \right), \quad \hat{B} := \rho \left( \delta \left( \begin{array}{c} \bullet \\ \bullet \end{array} \leftarrow \begin{array}{c} \bullet \\ \bullet \end{array} \hookrightarrow \begin{array}{c} \bullet \\ \bullet \end{array}; \neg \exists \left( \begin{array}{c} \bullet \\ \bullet \end{array} \hookrightarrow \begin{array}{c} \bullet \\ \bullet \end{array} \right) \right) \right) \\ \hat{C} &:= \rho \left( \delta \left( \begin{array}{c} \bullet \\ \bullet \end{array} \leftarrow \begin{array}{c} \bullet \\ \bullet \end{array} \hookrightarrow \begin{array}{c} \bullet \\ \bullet \end{array}; \text{true} \right) \right), \quad \hat{\mathcal{O}}(\hat{A}) = O_\bullet, \quad \hat{\mathcal{O}}(\hat{B}) = 2O_{\bullet|\bullet}, \quad \hat{\mathcal{O}}(\hat{C}) = 2O_{\bullet\bullet} \end{aligned}$$

Picking for simplicity as an initial state  $|\Psi(0)\rangle = |\emptyset\rangle$  just the empty graph, and invoking the SqPO-type jump-closure property (cf. Theorem 4) repeatedly in order to evaluate  $\langle [O_P, \hat{H}] \rangle(t) = \langle \hat{\mathcal{O}}([O_P, \hat{H}]) \rangle(t)$ , the moment EGF evolution equation (23) specializes to the following “Ehrenfest-like” [7] ODE system:

$$\begin{aligned} \frac{d}{dt} \langle O_\bullet \rangle(t) &= \langle [O_\bullet, H] \rangle(t) = \nu_+ - \nu_- \langle O_\bullet \rangle(t) \\ \frac{d}{dt} \langle O_{\bullet|\bullet} \rangle(t) &= \langle [O_{\bullet|\bullet}, H] \rangle(t) = \nu_+ \langle O_\bullet \rangle(t) - (2\nu_- + \varepsilon_+) \langle O_{\bullet|\bullet} \rangle(t) + \varepsilon_- \langle O_{\bullet\bullet} \rangle(t) \\ \frac{d}{dt} \langle O_{\bullet\bullet} \rangle(t) &= \langle [O_{\bullet\bullet}, H] \rangle(t) = \varepsilon_+ \langle O_{\bullet|\bullet} \rangle(t) - (2\nu_- + \varepsilon_-) \langle O_{\bullet\bullet} \rangle(t) \\ \langle O_\bullet \rangle(0) &= \langle O_{\bullet|\bullet} \rangle(0) = \langle O_{\bullet\bullet} \rangle(0) = 0. \end{aligned}$$

This ODE system may be solved exactly (see Appendix C). We depict in Figure 1 two exemplary evolutions of the three average pattern counts for different choices of parameters. Since due to SqPO-semantics the vertex deletion and creation transitions are entirely independent of the edge creation and deletion transitions, the vertex counts stabilize on a Poisson distribution of parameter  $\nu_+/\nu_-$  (where we only present the average vertex count value here). As for the non-linked vertex pair and edge pattern counts, the precise average values are sensitive to the parameter choices (i.e. whether or not vertices tend to be linked by an edge or not may be freely tuned in this model via adjusting the parameters).

While the example presented was chosen mainly to illustrate the computational techniques, it highlights the typical feature of the emergence of contributions in the relevant (nested) commutator calculations that may not have been included in the non-diagonal part  $\hat{H}$  of the infinitesimal generator of the CTMC. We refer the interested readers to [7] for an extended discussion of this phenomenon, and for computation strategies for higher-order moment evolution equations.



**Fig. 1.** Time-evolutions of pattern count observables for different parameter choices.

## 5 Application scenario 1: biochemistry with Kappa

The **KAPPA platform** [18,17] for rule-based modeling of biochemical reaction systems is based upon the notion of so-called *site-graphs* that abstract proteins and other complex macro-molecules into *agents* (with *sites* representing interaction capacities of the molecules). This open source platform offers a variety of high-performance *simulation algorithms* (for CTMCs based upon KAPPA rewriting rules) as well as several variants of static analysis tools to analyze and verify biochemical models [11]. In view of the present paper, it is interesting to note that since the start of the KAPPA development, the simulation-based algorithms have been augmented by *differential semantics* modules aimed at deriving ODE systems for the evolution of pattern-count observable average values [13,14,16,29]. In this section, we will experiment with a (re-)encoding of KAPPA in terms of typed undirected graphs with certain structural constraints that permits to express such moment statistics ODEs via our general rule-algebraic stochastic mechanics formalism. We will then provide an illustrative exemplary computation of ODEs in order to point out certain intrinsic intricacies (notably non-closure properties) typical of such calculations. One of the key theoretical features of KAPPA is its foundation upon the notion of *rigidity* [15]. In practice, the construction involves an *ambient category*  $\mathbb{A}$  (which possesses suitable adhesivity properties), a *pattern category*  $\mathbb{P}$  (obtained from  $\mathbb{A}$  via certain *negative constraints*) and finally a *state category*  $\mathbb{S}$  (obtained from  $\mathbb{P}$  via additional *positive constraints*). We will now present one possible realization of KAPPA based upon the  $\mathcal{M}$ -adhesive category of typed undirected multigraphs:

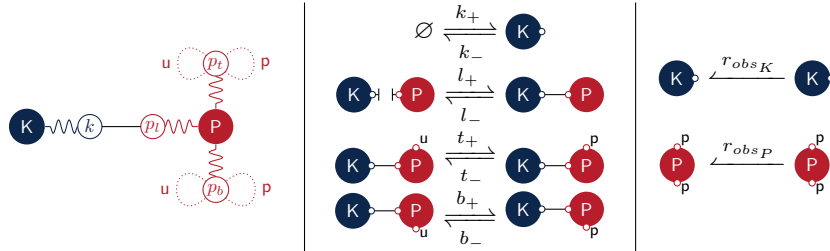
**Definition 8.** For a KAPPA model  $K$ , let  $\mathbb{A} = \mathbf{uGraph}/T_K$  be the category of finite undirected multigraphs typed over  $T_K$ , where  $T_K$  distinguishes agent vertex types, site vertex types and three forms of edge types: agent-site, site-site and loops on sites. For each agent type vertex  $X \in \{A, B, \dots\}$ , the type graph contains the site type vertices  $x_1 : X, \dots, x_{n_X} : X$  (incident to the  $X$ -type vertex via an edge, and where  $n_X < \infty$ ).  $T_K$  also contains link type edges between sites that encode which sites can be linked, and loops on site type vertices that represent

dynamic attributes, such as the phosphorylation state of a site. Indicating the three different edge types by wavy lines (agent-site), solid lines (site-site) and dotted lines (property loops), the agent vertices with filled circles  $\bullet$  and the site vertices by open circles  $\circ$ , and using the placeholder  $\bullet$  for a vertex and a dashed line for an edge of any type, we may introduce the negative constraints defining the pattern category  $\mathbb{P}_K$  as  $\mathcal{C}_{\mathcal{N}_K} := \bigwedge_{N \in \mathcal{N}_K} \neg \exists (\emptyset \hookrightarrow N)$ , with the set  $\mathcal{N}_K$  of “forbidden subgraphs” defined as

$$\mathcal{N}_K := \{\bullet \text{---} \bullet\} \cup \bigcup_x \left\{ \begin{array}{c} \circ \\ \circ \end{array} \right\} \cup \bigcup_{X,x} \left\{ \begin{array}{c} \circ \text{---} \bullet \\ \bullet \text{---} \circ \end{array} \right\}, \quad (27)$$

Finally, the state category  $\mathbb{S}_K$  is obtained from  $\mathbb{P}_K$  via imposing a positive constraint  $\mathcal{C}_{\mathcal{P}_K}$  that ensures that each agent  $X$  is linked to exactly one of each of its site vertices  $x : X$ , and if a site  $x : X$  can carry a property or alternative variants thereof, it also carries a loop that signifies one of these properties (see the example below for further details). Moreover, a given site  $x : X$  must be linked to an agent  $X$  (i.e. cannot occur in isolation).

*Example 3.* Consider a simple KAPPA model with a type graph as below left that introduces two agent types  $K$  (for “kinase”) and  $P$  (for “protein”), where  $K$  has a site  $k : K$ , and where  $P$  has sites  $p_t, p_l, p_b : P$ . Moreover, the sites  $p_t$  and  $p_b$  can carry properties  $u$  (“unphosphorylated”) and  $p$  (“phosphorylated”), depicted as dotted loops in the type graph. Sites  $k : K$  and  $p_l : P$  can bind (as indicated by the solid line in the type graph).



As a prototypical example of a KAPPA stochastic rewriting system, consider a system based upon the rewriting rules  $k_{\pm}$ ,  $l_{\pm}$ ,  $t_{\pm}$  and  $b_{\pm}$ . Here, for the rule  $l_+$ , we have indicated that it must be equipped with an application condition that ensures that the site of the  $K$ -type agent and the left site of the  $P$ -type agent must be *free* before binding. As common practice also in the standard KAPPA theory, we otherwise leave in the graphical depictions those application conditions necessary to ensure consistent matches implicit as much as possible. Consider then for a concrete computational example the time-evolution of the average count of the pattern described in the identity rule  $r_{obs_P}$ . As typical in KAPPA rule specifications  $r_{obs_P}$  as well as several of the other rules depicted only explicitly involve *patterns*, but not necessarily *states*, since e.g. in  $r_{obs_P}$  the left site of the  $P$ -type agent is not mentioned. In complete analogy to the

computation presented in Example 2, let us first compute the commutators of the observable  $O_K = \rho(\delta(r_{obs_K}; c_{obs_K}))$  with the operators  $\hat{X} := \rho(\delta(r_X; c_X))$ :

$$[O_K, \hat{K}_\pm] = \pm \hat{K}_\pm, [O_K, \hat{L}_\pm] = [O_K, \hat{T}_\pm] = [O_K, \hat{B}_\pm] = 0 \quad (28)$$

However, letting  $O_P^{(x,y)}$ ,  $O_{link}^{(x,y)}$  and  $O_{free}^{(x,y)}$  denote the observables for the patterns

$$\omega_P^{(x,y)} := \begin{array}{c} \text{x} \\ \circ \\ \text{P} \\ \circ \\ \text{y} \end{array}, \quad \omega_{link}^{(x,y)} := \begin{array}{c} \text{x} \\ \circ \\ \text{K} \text{---} \text{P} \\ \circ \\ \text{y} \end{array}, \quad \omega_{free}^{(x,y)} := \begin{array}{c} \text{x} \\ \circ \\ \text{K} \text{---} \text{K} \text{---} \text{P} \\ \circ \\ \text{y} \end{array}$$

one may easily demonstrate that even a comparatively simple observable such as  $O_P^{(p,p)}$  already leads to an infinite cascade of contributions to the ODEs for the averages of pattern counts. As typical in these sorts of computations, the discovery of a new pattern observable via applying SqPO-type jump-closure (Theorem 4) to the commutator contributions to  $\frac{d}{dt} \langle O_P^{(p,p)} \rangle(t)$  leads to the discovery of new pattern observables yet again, such as in

$$[O_P, \hat{T}_+] = \hat{T}_+^{(p)}, \quad \hat{\mathbb{O}}(\hat{T}_+^{(p)}) = O_{link}^{(u,p)}, \quad [O_{link}^{(u,p)}, \hat{L}_+] = \hat{L}_+^{(u,p)}, \quad \hat{\mathbb{O}}(\hat{L}_+^{(u,p)}) = O_{free}^{(u,p)}.$$

In particular the last observable  $O_{free}^{(u,p)}$  is found to lead to an infinite tower of other observables (i.e. “ODE non-closure”), starting from

$$[O_{free}^{(u,p)}, \hat{L}_+] = -\hat{L}_+^{(u,p)} - \left( \begin{array}{c} \text{K} \text{---} \text{P}^u \\ \text{K} \text{---} \text{K} \text{---} \text{P}^u \\ \text{K} \text{---} \end{array} \leftarrow \begin{array}{c} \text{K} \text{---} \text{P}^u \\ \text{K} \text{---} \text{K} \text{---} \text{P}^u \\ \text{K} \text{---} \end{array} \right) - \left( \begin{array}{c} \text{K} \text{---} \text{P} \\ \text{K} \text{---} \text{K} \text{---} \text{P} \\ \text{K} \text{---} \text{P}^u \end{array} \leftarrow \begin{array}{c} \text{K} \text{---} \text{P} \\ \text{K} \text{---} \text{K} \text{---} \text{P} \\ \text{K} \text{---} \text{P}^u \end{array} \right).$$

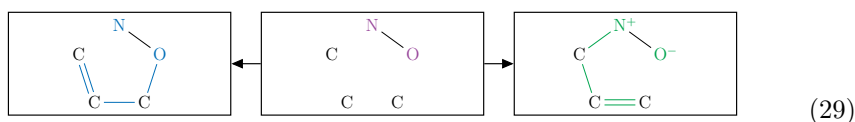
This exemplary and preliminary analysis reveals that while the rule-algebraic CTMC implementation is in principle applicable to the formulation and analysis KAPPA systems, further algorithmic and theoretical developments will be necessary (including possibly ideas of *fragments* and *refinements* as in [13,14,29]) in order to obtain a computationally useful alternative rewriting-theoretic implementation of KAPPA.

## 6 Application scenario 2: organic chemistry with MØD

The **MØD platform** [1] for organo-chemical reaction systems is a prominent example of a DPO-type rewriting theory of high relevance to the life sciences. From a theoretical perspective, MØD has been designed [3] as a rewriting system over so-called *chemical graphs*, a certain typed and undirected variant of the category **PLG** of partially labelled directed graphs. While the latter category had been introduced in [28] as a key example of an  $\mathcal{M}\mathcal{N}$ -adhesive category, with the motivation of permitting label-changes in rewriting rules, it was also demonstrated in loc cit. that **PLG** is *not*  $\mathcal{M}$ -adhesive. Since moreover no concrete construction of a tentative variant **uPLG** of **PLG** for undirected graphs, let alone results on the possible adhesivity properties of such a category are known in the literature,

we propose here an alternative and equivalent encoding of chemical graphs. We mirror the constructions of [1,3] in that chemical graphs will be a certain typed variant of undirected graphs, with *vertex types* representing *atom types*, *edge types* ranging over the types  $\{-, =, \#, :\}$  representing *single, double, triple and aromatic bonds*, respectively, and with the graphs being required to not contain multiedges. Inspired by the KAPPA constructions in the previous section, we opt to represent *properties* (such as e.g. *charges* on atoms) as *typed loop edges* on vertices representing atoms, whence the change of a property (which was the main motivation in [3] for utilizing a variant of **PLG**) may be encoded in a rewriting rule simply via deletion/creation of property-encoding loops. Unfortunately, while the heuristics presented thus far would suggest that chemical graphs in the alternative categorical setting should be just simple typed undirected graphs, the full specification of chemical graphs would also have to include additional, empirical information from the chemistry literature. Concretely, atoms such as e.g. carbon only support a limited variety of bond types and configurations of incident bonds (referred to as *valencies*), with additional complications such as poly-valencies possible for some types of atoms as illustrated by the following example.

*Example 4.* The *Meisenheimer-2-3-rearrangement* reaction [31] (cf. also [2]) constitutes an example<sup>5</sup> of a reaction where *polyvalence* is encountered:



Upon matching this rule into a chemically valid mixture, the *N* atom on the input of the rule will have valence 5, while on the output it will have valence 3. This type of information is evidently in no way contained in the chemical graphs alone, and must therefore be encoded in terms of suitable additional typing on the graphs and application conditions.

While thus at present no encoding of chemical graphs into a categorical framework with suitable adhesivity properties is available, we posit that it would be highly fruitful in light of the stochastic mechanics framework presented in this paper to develop such an encoding (joint work in progress with J.L. Andersen, W. Fontana and D. Merkle).

## 7 Conclusion and outlook

Rewriting theories of DPO- and SqPO-type for rules with conditions over  $\mathcal{M}$ -adhesive categories are poised to provide a rich theoretical and algorithmic framework for modeling stochastic dynamical systems in the life sciences. The main result of the present paper consists in the introduction of a *rule algebra*

<sup>5</sup> This example reaction was typeset directly via MØD (cf. Appendix D).



*framework* that extends the pre-existing constructions [9,4,6] precisely via incorporating the notion of conditions. The sophisticated KAPPA [11] and MØD [1] bio-/organo-chemistry platforms and related developments have posed one of the main motivations for this work. For both of these platforms, we present a first analysis and stepping stones towards bridging category-theoretical rewriting theories and stochastic mechanics computations. Especially for the organo-chemistry setting, our work motivates the development of a full encoding of (at least a reasonable fragment of) organic chemistry in terms of *chemical graphs* and rewriting rules thereof, which to date is still unavailable. This encoding will be beneficial also in the development of tracelet-based techniques [5], and is current work in progress.

An intriguing perspective for future developments in categorical rewriting theory consists in developing a robust and versatile methodology for the analysis of ODE systems of pattern-counting observables in stochastic rewriting systems. While the results of this paper permit to formulate dynamical evolution equations for arbitrary higher moments of such observables, in general cases (as illustrated in Section 5) the non-closure of the resulting ODE systems remains a fundamental technical challenge. In the KAPPA literature, sophisticated conceptual and algorithmic approaches to tackle this problem have been developed such as refinements [13,15], model reduction techniques [14] and stochastic fragments [25] (see also [7] for an extended discussion). We envision that a detailed understanding of these approaches from within the setting of categorical rewriting and of rule algebra theory could provide a very fruitful enrichment of the methodology of rewriting theory.

## References

1. Andersen, J.L., Flamm, C., Merkle, D., Stadler, P.F.: A Software Package for Chemically Inspired Graph Transformation. In: Graph Transformation, pp. 73–88. Springer International Publishing (2016). [https://doi.org/10.1007/978-3-319-40530-8\\_5](https://doi.org/10.1007/978-3-319-40530-8_5)
2. Andersen, J.L., Flamm, C., Merkle, D., Stadler, P.F.: An intermediate level of abstraction for computational systems chemistry. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences **375**(2109), 20160354 (2017). <https://doi.org/10.1098/rsta.2016.0354>
3. Andersen, J.L., Flamm, C., Merkle, D., Stadler, P.F.: Rule composition in graph transformation models of chemical reactions. Match **80**(3), 661–704 (2018)
4. Behr, N.: Sesqui-Pushout Rewriting: Concurrency, Associativity and Rule Algebra Framework. In: Echahed, R., Plump, D. (eds.) Proceedings of the Tenth International Workshop on Graph Computation Models (GCM 2019) in Eindhoven, The Netherlands. Electronic Proceedings in Theoretical Computer Science, vol. 309, pp. 23–52. Open Publishing Association (2019). <https://doi.org/10.4204/eptcs.309.2>
5. Behr, N.: Tracelets and Tracelet Analysis Of Compositional Rewriting Systems (accepted for ACT 2019 in Oxford). [arXiv:1904.12829](https://arxiv.org/abs/1904.12829) (2019)
6. Behr, N., Danos, V., Garnier, I.: Stochastic mechanics of graph rewriting. In: Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science - LICS '16. ACM Press (2016). <https://doi.org/10.1145/2933575.2934537>

7. Behr, N., Danos, V., Garnier, I.: Combinatorial Conversion and Moment Bisimulation for Stochastic Rewriting Systems. [arXiv:1904.07313](#) (2019)
8. Behr, N., Krivine, J.: Compositionality of Rewriting Rules with Conditions. [arXiv:1904.09322](#) (2019)
9. Behr, N., Sobocinski, P.: Rule Algebras for Adhesive Categories. In: Ghica, D., Jung, A. (eds.) 27th EACSL Annual Conference on Computer Science Logic (CSL 2018). Leibniz International Proceedings in Informatics (LIPIcs), vol. 119, pp. 11:1–11:21. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany (2018). <https://doi.org/10.4230/LIPIcs.CSL.2018.11>
10. Behr, N., Sobocinski, P.: Rule Algebras for Adhesive Categories (invited extended journal version). [arXiv:1807.00785](#) (2019)
11. Boutillier, P., Maasha, M., Li, X., Medina-Abarca, H.F., Krivine, J., Feret, J., Cristescu, I., Forbes, A.G., Fontana, W.: The kappa platform for rule-based modeling. *Bioinformatics* **34**(13), i583–i592 (2018). <https://doi.org/10.1093/bioinformatics/bty272>
12. Corradini, A., Heindel, T., Hermann, F., König, B.: Sesqui-Pushout Rewriting. In: Corradini, A., Ehrig, H., Montanari, U., Ribeiro, L., Rozenberg, G. (eds.) *Graph Transformations. Lecture Notes in Computer Science*, vol. 4178, pp. 30–45. Springer Berlin Heidelberg, Berlin, Heidelberg (2006)
13. Danos, V., Feret, J., Fontana, W., Harmer, R., Krivine, J.: Rule-based modelling, symmetries, refinements. In: Fisher, J. (ed.) *Formal Methods in Systems Biology*. pp. 103–122. Springer Berlin Heidelberg, Berlin, Heidelberg (2008). [https://doi.org/10.1007/978-3-540-68413-8\\_8](https://doi.org/10.1007/978-3-540-68413-8_8)
14. Danos, V., Feret, J., Fontana, W., Harmer, R., Krivine, J.: 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 (2010). <https://doi.org/10.1109/lics.2010.44>
15. Danos, V., Heckel, R., Sobocinski, P.: Transformation and Refinement of Rigid Structures. In: Giese H., K.B. (ed.) *Graph Transformation (ICGT 2014). Lecture Notes in Computer Science*, vol. 8571, pp. 146–160. Springer International Publishing (2014). [https://doi.org/10.1007/978-3-319-09108-2\\_10](https://doi.org/10.1007/978-3-319-09108-2_10)
16. Danos, V., Heindel, T., Honorato-Zimmer, R., Stucki, S.: Moment semantics for reversible rule-based systems. In: *International Conference on Reversible Computation*. pp. 3–26. Springer (2015). [https://doi.org/10.1007/978-3-319-20860-2\\_1](https://doi.org/10.1007/978-3-319-20860-2_1)
17. Danos, V., Laneve, C.: Formal molecular biology. *Theoretical Computer Science* **325**(1), 69–110 (2004). <https://doi.org/10.1016/j.tcs.2004.03.065>
18. Danos, V., Schachter, V.: Computational Methods in Systems Biology. In: *Conference proceedings CMSB*. p. 91. Springer (2004). <https://doi.org/10.1007/b107287>
19. Delbrück, M.: Statistical Fluctuations in Autocatalytic Reactions. *The Journal of Chemical Physics* **8**(1), 120–124 (1940). <https://doi.org/10.1063/1.1750549>
20. Ehrig, H., Ehrig, K., Prange, U., Taentzer, G.: *Fundamentals of Algebraic Graph Transformation. Monographs in Theoretical Computer Science (An EATCS Series)* (2006). <https://doi.org/10.1007/3-540-31188-2>
21. Ehrig, H., Golas, U., Habel, A., Lambers, L., Orejas, F.:  $\mathcal{M}$ -Adhesive Transformation Systems with Nested Application Conditions. Part 2: Embedding, Critical Pairs and Local Confluence. *Fundamenta Informaticae* **118**(1-2), 35–63 (2012). <https://doi.org/10.3233/FI-2012-705>
22. Ehrig, H., Golas, U., Habel, A., Lambers, L., Orejas, F.:  $\mathcal{M}$ -adhesive transformation systems with nested application conditions. Part 1: parallelism, concurrency and amalgamation. *Mathematical Structures in Computer Science* **24**(04) (2014). <https://doi.org/10.1017/s0960129512000357>

23. Ehrig, H., Golas, U., Hermann, F.: Categorical frameworks for graph transformation and HLR systems based on the DPO approach. *Bulletin of the EATCS* (102), 111–121 (2010)
24. Ehrig, H., Habel, A., Padberg, J., Prange, U.: Adhesive High-Level Replacement Categories and Systems. In: *Lecture Notes in Computer Science*, pp. 144–160. Springer Berlin Heidelberg (2004). [https://doi.org/10.1007/978-3-540-30203-2\\_12](https://doi.org/10.1007/978-3-540-30203-2_12)
25. Feret, J., Koepl, H., Petrov, T.: Stochastic fragments: A framework for the exact reduction of the stochastic semantics of rule-based models. *International Journal of Software and Informatics (IJSI)* **7**(4), 527–604 (2014)
26. Gabriel, K., Braatz, B., Ehrig, H., Golas, U.: Finitary  $\mathcal{M}$ -adhesive categories. *Mathematical Structures in Computer Science* **24**(04) (2014). <https://doi.org/10.1017/S0960129512000321>
27. Habel, A., Pennemann, K.H.: Correctness of high-level transformation systems relative to nested conditions. *Mathematical Structures in Computer Science* **19**(02), 245 (2009). <https://doi.org/10.1017/s0960129508007202>
28. Habel, A., Plump, D.:  $\mathcal{M}, \mathcal{N}$ -Adhesive Transformation Systems. In: Ehrig, H., Engels, G., Kreowski, H., Rozenberg, G. (eds.) *Graph Transformations (ICGT 2012)*. *Lecture Notes in Computer Science*, vol. 7562, pp. 218–233. Springer Berlin Heidelberg (2012). [https://doi.org/10.1007/978-3-642-33654-6\\_15](https://doi.org/10.1007/978-3-642-33654-6_15)
29. Harmer, R., Danos, V., Feret, J., Krivine, J., Fontana, W.: Intrinsic information carriers in combinatorial dynamical systems. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **20**(3), 037108 (2010). <https://doi.org/10.1063/1.3491100>
30. Lack, S., Sobociński, P.: Adhesive and quasiadhesive categories. *RAIRO - Theoretical Informatics and Applications* **39**(3), 511–545 (2005). <https://doi.org/10.1051/ita:2005028>
31. March, J.: *March’s Advanced Organic Chemistry: Reactions, Mechanisms and Structure*, vol. 5. American Chemical Society (ACS) (2001). <https://doi.org/10.1021/op010074a>
32. Norris, J.R.: *Markov Chains*. Cambridge University Press (1997). <https://doi.org/10.1017/cbo9780511810633>
33. Padberg, J.: Towards M-Adhesive Categories based on Coalgebras and Comma Categories. [arXiv:1702.04650](https://arxiv.org/abs/1702.04650) (2017)

## A Background material on adhesive categories and rewriting with conditions

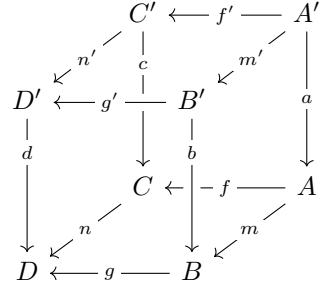
As a reference for notational conventions and in order to recall some of the standard definitions necessary in the main text, we collect here some of the materials contained in our recent paper [8] for the readers' convenience.

### A.1 $\mathcal{M}$ -adhesive categories

**Definition 9.** An  $\mathcal{M}$ -adhesive category [23]  $(\mathbf{C}, \mathcal{M})$  is a category  $\mathbf{C}$  together with a class of monomorphisms  $\mathcal{M}$  that satisfies the following properties:

1.  $\mathbf{C}$  has pushouts and pullbacks along<sup>6</sup>  $\mathcal{M}$ -morphisms.
2. The class  $\mathcal{M}$  contains all isomorphisms and is stable under pushout, pullback and composition.
3. Pushouts along  $\mathcal{M}$ -morphisms are  $\mathcal{M}$ -van Kampen squares.

The latter property entails that in a commutative diagram such as the one on the right where the bottom square is a pushout along an  $\mathcal{M}$ -morphism, where the back and right faces pullbacks and where all vertical morphisms are in  $\mathcal{M}$ , the bottom square is  $\mathcal{M}$ -van Kampen if the following property holds: the top square is a pushout if and only if the front and left squares are pullbacks.



Throughout the following definitions, let  $(\mathbf{C}, \mathcal{M})$  be an  $\mathcal{M}$ -adhesive category.

**Definition 10.**  $(\mathbf{C}, \mathcal{M})$  is said to be finitary [26] if every object has only finitely many  $\mathcal{M}$ -subobjects up to isomorphism.

**Definition 11.**  $(\mathbf{C}, \mathcal{M})$  possesses an  $\mathcal{M}$ -initial object  $\emptyset$  [26] if for all objects  $X \in \text{obj}(\mathbf{C})$  there exists a unique  $\mathcal{M}$ -morphism  $\iota_X : \emptyset \hookrightarrow X$ .

**Definition 12.**  $(\mathbf{C}, \mathcal{M})$  possesses an epi- $\mathcal{M}$ -factorization [27] if every morphism  $f \in \text{mor}(\mathbf{C})$  factorizes as  $f = m \circ e$  with  $m \in \mathcal{M}$  and with  $e \in \text{epi}(\mathbf{C})$  an epimorphism, and such that this factorization is unique up to isomorphism.

**Definition 13.**  $(\mathbf{C}, \mathcal{M})$  has  $\mathcal{M}$ -effective unions if for every cospan  $(B \hookrightarrow D \leftarrow C)$  of  $\mathcal{M}$ -morphisms that is the pushout of a span  $(B \leftarrow A \hookrightarrow C)$ , the following property holds: for every cospan  $(B \hookrightarrow E \leftarrow C)$  whose pullback is given by  $(B \leftarrow A \hookrightarrow C)$ , the morphism  $D \rightarrow E$  that exists by universal property of the pushout is in  $\mathcal{M}$ .

<sup>6</sup> Here, “along” entails that at least one of the two morphisms involved in the relevant (co-)span is in  $\mathcal{M}$ .

We next recall the notion of final pullback complements that is an important technical ingredient of the theory of SqPO-rewriting.

**Definition 14.** Let  $(b, a)$  be a composable pair of morphisms in a category  $\mathbf{C}$ . Then a pair of morphisms  $(c, d)$  is called a final pullback complement (FPC) [12] if  $(a, d)$  is the pullback of  $(b, c)$ , and if for every  $(a \circ p, q)$  that is the pullback of  $(b, r)$ , there exists a morphism  $s$  such that  $r = c \circ s$  that is unique up to isomorphism.

$$\begin{array}{ccccc}
 & & a \circ p & & \\
 & \swarrow & & \searrow & \\
 B & \xleftarrow{a} & A & \xleftarrow{p} & P \\
 b \downarrow & & d \downarrow & & q \downarrow \\
 C & \xleftarrow{c} & D & \xleftarrow{\exists s} & Q \\
 & \nwarrow & & \nearrow & \\
 & & r & & 
 \end{array}$$

**Definition 15.** The class of monomorphisms  $\mathcal{M}$  of  $(\mathbf{C}, \mathcal{M})$  is said to be stable under FPCs [8] if for every pair  $(b, a)$  of composable  $\mathcal{M}$ -morphisms the FPC  $(c, d)$  (if it exists) is a pair of  $\mathcal{M}$ -morphisms.

## A.2 Conditions

**Definition 16.** Conditions [27, 22] in an  $\mathcal{M}$ -adhesive category  $(\mathbf{C}, \mathcal{M})$  satisfying Assumption 1 are recursively defined for every object  $X \in \text{obj}(\mathbf{C})$  as follows:

1.  $\text{true}_X$  is a condition.
2. Given  $(f : X \hookrightarrow Y) \in \mathcal{M}$  and a condition  $c_Y$ ,  $\exists(f, c_Y)$  is a condition.
3. If  $c_X$  is a condition, so is  $\neg c_X$ .
4. If  $c_X^{(1)}, c_X^{(2)}$  are conditions, so is  $c_X^{(1)} \wedge c_X^{(2)}$ .

The satisfaction of a condition  $c_X$  by a  $\mathcal{M}$ -morphism  $(h : X \hookrightarrow Z) \in \mathcal{M}$ , denoted  $h \models c_X$ , is recursively defined (with notations as above) as follows:

1.  $h \models \text{true}_X$ .
2.  $h \models \exists(f, c_Y)$  iff there exists an  $\mathcal{M}$ -morphism  $(g : Y \hookrightarrow Z) \in \mathcal{M}$  such that  $h = g \circ f$  and  $g \models c_Y$ .
3.  $h \models \neg c_X$  iff  $h \not\models c_X$ .
4.  $h \models (c_X^{(1)} \wedge c_X^{(2)})$  iff  $h \models c_X^{(1)}$  and  $h \models c_X^{(2)}$ .

$$\begin{array}{ccc}
 \exists(f, c_Y) \nabla & & \nabla c_Y \\
 X & \xrightarrow{f} & Y \\
 & \searrow h & \swarrow g \\
 & & Z
 \end{array}$$

Two conditions  $c_X$  and  $c'_X$  are equivalent, denoted  $c_X \equiv c'_X$ , iff for every  $\mathcal{M}$ -morphism  $(h : X \hookrightarrow Z) \in \mathcal{M}$ ,  $h \models c_X$  if and only if  $h \models c'_X$ .

Finally, a condition  $c_\emptyset$  over the  $\mathcal{M}$ -initial object  $\emptyset$  is called a constraint, and we define for every object  $Z \in \text{obj}(\mathbf{C})$

$$Z \models c_\emptyset \quad :\Leftrightarrow \quad (\emptyset \hookrightarrow Z) \models c_\emptyset. \quad (30)$$

**Theorem 6** ([27]; cf. also [8]). *In an  $\mathcal{M}$ -adhesive category satisfying Assumption 1, there exists a shift operation, denoted  $\text{Shift}$ , such that for all conditions  $c_X$  and for all  $\mathcal{M}$ -morphisms  $(f : X \hookrightarrow Y) \in \mathcal{M}$ ,  $(g : Y \hookrightarrow Z) \in \mathcal{M}$  and  $(h : X \hookrightarrow Z) \in \mathcal{M}$  with  $h = g \circ f$ , the following property holds:*

$$h \models c_X \quad \Leftrightarrow \quad g \models \text{Shift}(f, c_X) \quad (31)$$

We refer the interested readers to [8] for further details on the concrete implementation of the shift construction.

### A.3 Associativity and concurrency theorems

In the statements of the following two theorems, we always imply choosing concrete representatives of the relevant equivalence classes of rules with conditions in order to list the sets of admissible matches.

**Theorem 7 (Associativity of rule compositions [9,4,8]).** *Let  $\mathbf{C}$  be a category satisfying Assumption 1. let  $R_1, R_2, R_3 \in \overline{\text{Lin}}(\mathbf{C})$  be linear rules with conditions, and let  $\mathbb{T} \in \{\text{DPO}, \text{SqPO}\}$ . Then there exists a bijection  $\varphi : A \xrightarrow{\cong} B$  of sets of pairs of  $\mathbb{T}$ -admissible matches  $A$  and  $B$ , defined as*

$$\begin{aligned} A &:= \{(\mu_{21}, \mu_{3(21)}) \mid \mu_{21} \in \mathbf{M}_{R_2}^{\mathbb{T}}(R_1), \mu_{3(21)} \in \mathbf{M}_{R_3}^{\mathbb{T}}(R_2)\} \\ B &:= \{(\mu_{32}, \mu_{(32)1}) \mid \mu_{32} \in \mathbf{M}_{R_3}^{\mathbb{T}}(R_2), \mu_{(32)1} \in \mathbf{M}_{R_{32}}^{\mathbb{T}}(R_1)\}, \end{aligned} \quad (32)$$

where  $R_{21} = R_2^{\mu_{21} \triangleleft_{\mathbb{T}}} R_1$  and  $R_{32} = R_3^{\mu_{32} \triangleleft_{\mathbb{T}}} R_2$ , such that for each corresponding pair  $(\mu_{21}, \mu_{3(21)}) \in A$  and  $\varphi(\mu_{21}, \mu_{3(21)}) = (\mu'_{32}, \mu'_{(32)1}) \in B$ ,

$$R_3^{\mu_{3(21)} \triangleleft_{\mathbb{T}}} (R_2^{\mu_{21} \triangleleft_{\mathbb{T}}} R_1) \cong (R_3^{\mu'_{32} \triangleleft_{\mathbb{T}}} R_2)^{\mu'_{(32)1} \triangleleft_{\mathbb{T}}} R_1. \quad (33)$$

In this particular sense, the composition operations  $\triangleleft_{\mathbb{T}}$  are **associative**.

**Theorem 8 (Concurrency theorem [9,4,8]).** *Let  $\mathbf{C}$  be a category satisfying Assumption 1, and let  $\mathbb{T} \in \{\text{DPO}, \text{SqPO}\}$ . Then there exists a bijection  $\varphi : A \xrightarrow{\cong} B$  on pairs of  $\mathbb{T}$ -admissible matches between the sets  $A$  and  $B$ ,*

$$\begin{aligned} A &= \{(m_2, m_1) \mid m_1 \in \mathbf{M}_{R_1}^{\mathbb{T}}(X_0), m_2 \in \mathbf{M}_{R_2}^{\mathbb{T}}(X_1)\} \\ &\cong B = \{(\mu_{21}, m_{21}) \mid \mu_{21} \in \mathbf{M}_{R_2}^{\mathbb{T}}(R_1), m_{21} \in \mathbf{M}_{R_{21}}^{\mathbb{T}}(X_0)\}, \end{aligned} \quad (34)$$

where  $X_1 = R_{1_{m_1}}(X_0)$  and  $R_{21} = R_2^{\mu_{21} \triangleleft_{\mathbb{T}}} R_1$  such that for each corresponding pair  $(m_2, m_1) \in A$  and  $(\mu_{21}, m_{21}) \in B$ , it holds that

$$R_{21_{m_{21}}}(X_0) \cong R_{2_{m_2}}(R_{1_{m_1}}(X_0)). \quad (35)$$

## B Proofs

### B.1 Proof of Theorem 3

The statement of the theorem is equivalent to the following two properties:

- (i)  $\bar{\rho}_{\mathbf{C}}^{\mathbb{T}}(\delta(R_{\emptyset})) = Id_{End_{\mathbb{R}}(\hat{\mathbf{C}})}$
- (ii)  $\forall R_1, R_2 \in \overline{\text{Lin}}(\mathbf{C})_{\sim} : \bar{\rho}_{\mathbf{C}}^{\mathbb{T}}(\delta(R_2)) \bar{\rho}_{\mathbf{C}}^{\mathbb{T}}(\delta(R_1)) = \bar{\rho}_{\mathbf{C}}^{\mathbb{T}}(\delta(R_2) \star_{\mathbb{T}} \delta(R_1)) .$

By linearity, it suffices to verify these properties on an arbitrary basis vector  $|X\rangle \in \hat{\mathbf{C}}$ . For (i), it suffices to verify that

$$\bar{\rho}_{\mathbf{C}}^{\mathbb{T}}(\delta(R_{\emptyset})) |X\rangle = \sum_{m \in \mathbb{M}_{R_{\emptyset}}^{\mathbb{T}}(X)} |R_{\emptyset m}(X)\rangle = |X\rangle .$$

Property (ii) is a consequence of Theorem 8 (the Concurrency Theorem):

$$\begin{aligned} \bar{\rho}_{\mathbf{C}}^{\mathbb{T}}(\delta(R_2)) \bar{\rho}_{\mathbf{C}}^{\mathbb{T}}(\delta(R_1)) |X\rangle &= \sum_{m_1 \in \mathbb{M}_{R_1}^{\mathbb{T}}(X)} \sum_{m_2 \in \mathbb{M}_{R_2}^{\mathbb{T}}(R_{1m_1}(X))} |R_{2m_2}(R_{1m_1}(X))\rangle \\ &= \sum_{\mu \in \mathbb{M}_{R_2}^{\mathbb{T}}(R_1)} \sum_{m_{21} \in \mathbb{M}_{R_{2\mu}^{\mathbb{T}}}(X)} |R_{2\mu 1 m_{21}}(X)\rangle . \end{aligned}$$

### B.2 Proof of Theorem 4

*Ad 1.:* It suffices to verify that direct derivations along a rule  $R$  of the relevant form occurring in the two types of observables from any object  $X$  satisfy  $R_m(X) \cong X$ . But this follows directly from the respective definitions of direct derivations.

*Ad 2. & 3.:* It again suffices to verify these properties on basis elements  $|X\rangle$  of  $\hat{\mathbf{C}}$ , and for generic  $R \in \overline{\text{Lin}}(\mathbf{C})_{\sim}$ . By definition,

$$\langle | \bar{\rho}_{\mathbf{C}}^{\mathbb{T}}(\delta(R)) |X\rangle = \sum_{m \in \mathbb{M}_R^{\mathbb{T}}(X)} \underbrace{\langle | R_m(X) \rangle}_{=1_{\mathbb{R}}} = |\mathbb{M}_R^{\mathbb{T}}(X)| . \quad (36)$$

In both cases of semantics, a candidate match of  $R$  into  $X$  must satisfy the application condition. In the DPO case, in addition the relevant pushout complement must exist. Combining these facts allows to verify the formulae for  $\hat{\mathbb{O}}(\cdot)$ .

*Ad 4.:* The proof is straightforward generalization of the corresponding statement for the case of rewriting rules without conditions [10, 4]. Following standard continuous-time Markov chain (CTMC) theory [32], one may verify that the linear operator  $\mathcal{H}$  has a strictly negative coefficient diagonal contribution  $\hat{\mathbb{O}}(H)$ , a non-negative coefficient off-diagonal contribution  $H$ , thus  $\mathcal{H}$  satisfies  $\langle | \mathcal{H} = 0$ . Since in addition a given  $X \in \text{obj}(\mathbf{C})_{\cong}$  may be rewritten via direct derivations along the rules of the transition set only in finitely many ways, in summary  $\mathcal{H}$  fulfills all requirements to qualify as a conservative and stable  $Q$ -matrix (i.e. an infinitesimal generator) of a CTMC (cf. [4] for further details).

## C Details on the symbolic solution to the observable average counts in Example 2

The ODE system of Example 2 may be solved in closed form as follows:

$$\begin{aligned}
 \langle O_{\bullet} \rangle(t) &= \frac{\nu_+}{\nu_-} (1 - e^{-t\nu_-}) \\
 \langle O_{\bullet|\bullet} \rangle(t) &= \frac{\nu_+^2 e^{-\alpha t}}{2\alpha\beta\lambda\nu_-^2} (\alpha\beta\varepsilon_- e^{\lambda t} + 2\varepsilon_+\nu_-^2 - 2\alpha\kappa\lambda e^{\beta t} + \beta\lambda\omega e^{\alpha t}) \\
 \langle O_{\bullet\bullet} \rangle(t) &= \frac{\varepsilon_+\nu_+^2 e^{-\alpha t}}{2\alpha\beta\lambda\nu_-^2} (\alpha\beta e^{\lambda t} - 2\alpha\lambda e^{\beta t} + \beta\lambda e^{\alpha t} - 2\nu_-^2) \\
 \alpha &= \varepsilon_- + \varepsilon_+ + 2\nu_- , \quad \beta = \varepsilon_- + \varepsilon_+ + \nu_- \\
 \kappa &= \varepsilon_- + \nu_- , \quad \lambda = \varepsilon_- + \varepsilon_+ , \quad \omega = \varepsilon_- + 2\nu_- .
 \end{aligned} \tag{37}$$

In particular, one may provide asymptotic formulae for  $t \rightarrow \infty$ :

$$\begin{aligned}
 \langle O_{\bullet} \rangle(t) &\xrightarrow{t \rightarrow \infty} \frac{\nu_+}{\nu_-} \\
 \langle O_{\bullet|\bullet} \rangle(t) &\xrightarrow{t \rightarrow \infty} \frac{\nu_+^2 (\varepsilon_- + 2\nu_-)}{2\nu_-^2 (\varepsilon_- + \varepsilon_+ + 2\nu_-)} \\
 \langle O_{\bullet\bullet} \rangle(t) &\xrightarrow{t \rightarrow \infty} \frac{\varepsilon_+\nu_+^2}{2\nu_-^2 (\varepsilon_- + \varepsilon_+ + 2\nu_-)} .
 \end{aligned} \tag{38}$$

## D Technical details of typesetting the MØD example

For the interested readers, the following code may be used in either a standalone instance or via the [live playground](#) of MØD [1] in order to reproduce the graphics for the Meisenheimer-2-3-rearrangement example of a organo-chemical reaction given in the main text. Note that since MØD employs the traditional “left-to-right” convention for rules, the input and output patterns are given as “right” and “left”, respectively.

```

# Meisenheimer-2-3 rearrangement:
meisenheimer = ruleGMLString("""rule [
  left [
    edge [ source 1 target 2 label "-" ]
    edge [ source 2 target 3 label "=" ]
    edge [ source 1 target 5 label "-" ]
    node [ id 4 label "N" ]
    node [ id 5 label "O" ]
  ]
  context [
    node [ id 1 label "C" ]
    node [ id 2 label "C" ]
    node [ id 3 label "C" ]
    edge [ source 4 target 5 label "-" ]
  ]
  right [
    edge [ source 1 target 2 label "=" ]

```



```
    edge [ source 2 target 3 label "-" ]  
    edge [ source 3 target 4 label "-" ]  
    node [ id 4 label "N+" ]  
    node [ id 5 label "O-" ]  
  ]  
] """)  
# Printing of the rule:  
meisenheimer.print()
```