A Categorical Model for Organic Chemistry

Ella Gale[a], Leo Lobski[b,*], Fabio Zanasi[b]

[a] *University of Bristol, Queens Road BS8 1QU, Bristol, UK*
[b] *University College London, Gower Street WC1E 6BT, London, UK*

Abstract

We introduce a mathematical framework for organic chemistry, with three interrelated perspectives on chemical processes: reaction schemes, formal reactions and disconnection rules. We apply the framework to retrosynthetic analysis, an important research method in synthetic chemistry. Our approach represents molecules as labelled graphs, while the interactions between them are represented either as double pushout graph rewriting, partial bijections or local edge rewrite rules. In particular, we show that the formal reactions are generated by reaction schemes using double pushout rewriting, and that the disconnection rules are sound, complete and universal with respect to chemical reactions. The mathematical formulation of retrosynthesis is based on layered props – a recently introduced categorical model for partial explanations in scientific reasoning.

Keywords: Chemical graph rewriting, Chemical reactions, Disconnection rules, Retrosynthesis

1. Introduction

A chemical reaction can be understood as a rule which tells us what the outcome molecules (or molecule-like objects, such as ions) are when several molecules are put together. If, moreover, the reaction records the precise proportions of the molecules as well as the conditions for the reaction to take place (temperature, pressure, concentration, presence of a solvent etc.), it can be seen as a precise scientific prediction, whose truth or falsity can be tested in a lab, making the reaction reproducible. Producing complicated molecules, as required e.g. by the pharmaceutical industry, requires, in general, a chain of several consecutive reactions in precisely specified conditions. The general task of synthetic chemistry is to come up with reproducible reaction chains to generate previously unknown molecules (with some desired properties) [1]. Successfully achieving a given synthetic task requires both understanding of the chemical mechanisms and the empirical knowledge of existing reactions. Both of these are increasingly supported by computational methods [2]: rule-based and dynamical models are used to suggest potential reaction mechanisms, while database search is used to look for existing reactions that would apply in the context of interest [3]. The key desiderata for such tools are tunability and

*Corresponding author
Email addresses: ella.gale@bristol.ac.uk (Ella Gale), leo.lobski.21@ucl.ac.uk (Leo Lobski), f.zanasi@ucl.ac.uk (Fabio Zanasi)

specificity. Tunability endows a synthetic chemist with tools to specify a set of goals (e.g. adding or removing a functional group[1]), while by specificity we mean maximising yield and minimising side products.

Retrosynthetic analysis. In this article, we focus on the area of synthetic chemistry known as *retrosynthesis* [4, 3, 5]. While reaction prediction asks what reactions will occur and what outcomes will be obtained when some molecules are allowed to interact, retrosynthesis goes backwards: it starts with a target molecule that we wish to produce, and it proceeds in the "reverse" direction by asking what potential reactants would produce the target molecule. While many automated tools for retrosynthesis exist (see e.g. [6, 7, 8, 9, 10, 11, 12]), there is no uniform mathematical grounding for compositional reasoning about retrosynthesis. The primary contribution of this paper is to provide such a mathematically sound framework relevant for the retrosynthetic practice. Indeed, all three kinds of transformations of chemical graphs we consider (reactions, reaction schemes and disconnection rules) appear in the automated retrosynthetic tools. By formalising the methodology at this level of mathematical generality, we are able to provide insights into incorporating features that the current automated retrosynthesis tools lack: these include modelling chirality, the reaction environment, and the protection-deprotection steps (see for example [13]), which are all highly relevant to practical applications. Our formalism, therefore, paves the way for new automated retrosynthesis tools, accounting for the aforementioned features.

Mathematically, our approach to retrosynthesis is phrased in the algebraic formalism of *string diagrams* [14], and most specifically uses *layered props*. Layered props were originally introduced, in [15], as models for systems that have several interdependent levels of description. In the context of chemistry, the description levels play a threefold role: first, each level represents a reaction environment, second, the morphisms in different levels are taking care of different synthetic tasks, and third, the rules that are available in a given level reflect the structure that is deemed relevant for the next retrosynthetic step. The latter can be seen as a kind of coarse-graining, where by deliberately restricting to a subset of all available information, we reveal some essential features about the system. Additionally, organising chemical processes into levels allows us to include conditions that certain parts of a molecule are to be kept intact. While the presentation here is self-contained and, in particular, does not assume a background on layered props, we emphasise that our approach is principled in the sense that many choices we make are suggested by this more general framework. We point such choices out when we feel the intuition that comes from layered props is helpful for understanding the formalism presented in the present work.

Threefold view of chemical graphs. Throughout the article, we take three perspectives on chemical processes (Figure 1), and discuss the ways in which they are interlinked. The first perspective is that of *reaction schemes* (Section 5), which encode how bonds and charges change when two parts of chemical compounds interact. Reaction schemes give rise to *reactions* via *double pushout graph rewriting* (Section 4). The category of formal reactions (Definition 5.7) is

[1]Part of a molecule that is known to be responsible for certain chemical function.

our second perspective: reactions can be thought of as combinatorial rearrangements of molecules that preserve matter and charge. The third perspective are the local graph rewrites used in retrosynthesis – known as *disconnection rules* – which capture any possible local change in charge or connectivity. The disconnection rules can be seen as a subset of reaction schemes, and as an axiomatisation of reactions: while they provide a fine-grained view of the chemical transformations, they allow us to recursively define a functor to reactions.

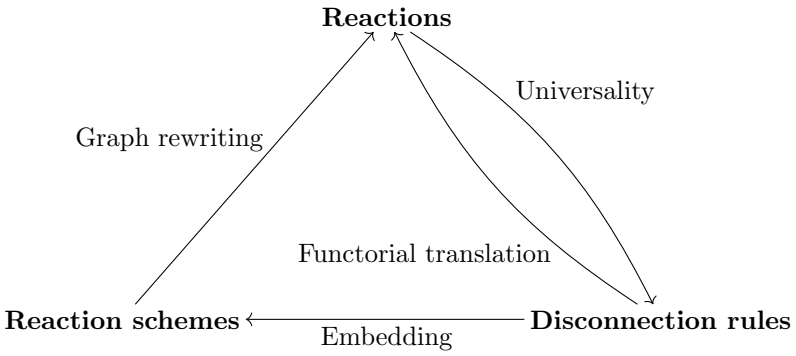


Figure 1: The three perspectives on chemical processes

Whereas chemical reactions have been studied formally before, a mathematical description of disconnection rules has received far less attention [16, 17, 18]. Our approach takes a novel perspective on the basic units of retrosynthetic analysis – the disconnection rules – by making them first-class citizens of reaction representation. The mathematical and conceptual justification for doing so lies in the fact that, as we show, both disconnection rules and reactions can be arranged into (monoidal) categories [17], such that there is a functor taking each sequence of disconnection rules to a reaction. Our main result concerning the disconnection rules states that, under a certain axiomatisation, the functor is faithful and full up to isomorphism. Such a categorical perspective provides a precise mathematical meaning to the claim that disconnection rules are sound, complete and universal with respect to the reactions. This implies that every reaction can be decomposed into a sequence of disconnection rules (universality) in an essentially unique way (completeness). More broadly, our contribution incorporates disconnection rules within the framework of applied category theory [19], which emphasises compositional modelling as a means to uniformly study systems across various disciplines of science.

Contributions. We list the novel contributions per section.

- in Section 4, we introduce morphisms of graphs representing molecules as functions that do not remove existing charge or matter, and show that the resulting category is $(\mathcal{M}, \mathcal{N})$-adhesive (Theorem 4.24),
- in Section 5, we define the category of reactions (Definition 5.7), and show that its morphisms arise as double pushout diagrams of reaction schemes (Theorem 5.5, Proposition 5.6),
- in Section 6, we introduce the mathematical description of retrosynthetic disconnection rules (Definition 6.1) together with equations (Definition 6.5) between them,

- Section 7 constructs a translation from the disconnection rules to reactions, and proves that it is complete (Theorem 7.3) and universal (Theorem 7.4),
- Section 9 introduces the mathematical description of retrosynthesis.

Synopsis. In Section 2, we give a brief overview of the methodology of retrosynthetic analysis, as well as of the existing tools for automating it. The entirety of Section 3 is devoted to constructing the labelled graphs that we use to represent molecular entities: these form the objects of all three categories that are discussed in the three following sections. Section 4 focusses on the categorical constructions needed for representing reactions as double pushout graph rewriting, which are used in Section 5 to define reactions and reaction schemes. Section 6 formalises retrosynthetic disconnection rules, from which we define a functor to reactions in Section 7, where it is moreover proved to be faithful and full up to isomorphism. Section 8 recalls the conceptual and mathematical ideas behind layered props. All the perspectives on chemical reactions come together in the layered prop defined in Section 9, where we also describe how to reason about retrosynthesis within it. In Section 10 we sketch the prospects of future work.

This article extends the conference paper *A Categorical Approach to Synthetic Chemistry* [17] with new material in Sections 4-7 and 9. Specifically, the proof of adhesivity (Section 4) and the results on universality and completeness (Section 7) are new. We have also included more detailed constructions, all the missing proofs, and several additional examples to illustrate the definitions. The material in Sections 6, 7 and Appendix A is based on a conference paper *Disconnection Rules are Complete for Chemical Reactions* [18], to appear in the proceedings of ICTAC 2024.

2. Retrosynthetic Analysis

Retrosynthetic analysis starts with a target molecule we wish to produce but do not know how. The aim is to "reduce" the target molecule to known (commercially available) outcome molecules in such a way that when the outcome molecules react, the target molecule is obtained as a product. This is done by (formally) partitioning the target molecule into functional parts referred to as *synthons*, and finding actually existing molecules that are chemically equivalent to the synthons; these are referred to as *synthetic equivalents* [20, 1, 21]. If no synthetic equivalents can be found that actually exist, the partitioning step can be repeated, this time using the synthetic equivalents themselves as the target molecules, and the process can continue until either known molecules are found, or a maximum number of steps is reached and the search is stopped. Note that the synthons themselves do not refer to any molecule as such, but are rather a convenient formal notation for parts of a molecule. For this reason, passing from synthons to synthetic equivalents is a non-trivial step involving intelligent guesswork and chemical know-how of how the synthons *would* react if they were independent chemical entities.

Clayden, Warren and Greeves [21] give the example in Figure 2 when introducing retrosynthesis (a synthesis of benzyl benzoate). Here the molecule on the left-hand side (benzyl benzoate) is the target, while the resulting two

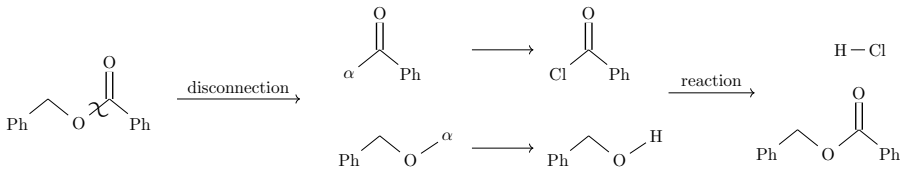


Figure 2: A retrosynthetic sequence for benzyl benzoate

parts with the symbol α are the synthons. We use the symbol α to indicate where the cut has been made, and hence which atoms have unpaired electrons. Replacing the symbols α in the synthons with `Cl` and `H`, we obtain the candidate synthetic equivalents (benzoyl chloride and benzyl alcohol) shown one step further to the right. As the next step, we find a reaction using the synthetic equivalents as reactants and having the target amongst the products (this can be done by e.g. looking up a reaction database). This is the simplest possible instance of a retrosynthetic sequence, in the sense that a reaction pathway is found in one iteration. In general, the interesting sequences are much longer, and, importantly, contain information under what conditions the reactions will take place.

2.1. Existing tools

Many tools for automatic retrosynthesis have been successfully developed starting from the 1960s [6, 7, 9, 10, 11]. They can be divided into two classes [3]: *template-based* [22, 23] and *template-free* [9, 24]. Template-based tools contain a rule database (the *template*), which is either manually encoded or automatically extracted. Given a molecule represented as a graph, the model checks whether any rules are applicable to it by going through the database and comparing the conditions of applying the rule to the subgraphs of the molecule [3]. Choosing the order in which the rules from the template and the subgraphs are tried are part of the model design. Template-free tools, on the other hand, are data-driven and treat the retrosynthetic rule application as a translation between graphs or their representations as strings: the suggested transforms are based on learning from known transforms, avoiding the need for a database of rules [3, 11].

While successful retrosynthesic sequences have been predicted by the computational retrosynthesis tools, they lack a rigorous mathematical foundation, which makes them difficult to compare, combine or modify. Other common drawbacks of the existing approaches include not including the reaction conditions or all cases of chirality as part of the reaction template [3, 9], as well as the fact that the existing models are unlikely to suggest protection-deprotection steps. Additionally, the template-free tools based on machine learning techniques sometimes produce output that does not correspond to molecules in any obvious way, and tend to reproduce the biases present in the literature or a data set [3].

For successful prediction, the reaction conditions are, of course, crucial. These include such factors as temperature and pressure, the presence of a solvent (a compound which takes part in the reaction and whose supply is essentially unbounded), the presence of a reagent (a compound without which the reaction would not occur, but which is not the main focus or the target), as well as the presence of a catalyst (a compound which increases the rate at which

the reaction occurs, but is itself unaltered by the reaction). The above factors can change the outcome of a reaction dramatically [25, 26]. There have indeed been several attempts to include reaction conditions into the forward reaction prediction models [27, 28, 29, 30]. However, the search space in retrosynthesis is already so large that adding another search criterion should be done with caution. A major challenge for predicting reaction conditions is that they tend to be reported incompletely or inconsistently in the reaction databases [31].

Chirality (mirror-image asymmetry) of a molecule can alter its chemical and physiological properties, and hence constitutes a major part of chemical information pertaining to a molecule. While template-based methods have been able to successfully suggest reactions involving chirality (e.g. [7]), the template-free models have difficulties handling it [9]. This further emphasises usefulness of a framework which is able to handle both template-based and template-free models.

The protection-deprotection steps are needed when more than one functional group of a molecule A would react with a molecule B. To ensure the desired reaction, the undesired functional group of A is first "protected" by adding a molecule X, which guarantees that the reaction product will react with B in the required way. Finally, the protected group is "deprotected", producing the desired outcome of B reacting with the correct functional group of A. So, instead of having a direct reaction $A + B \to C$ (which would not happen, or would happen imperfectly, due to a "competing" functional group), the reaction chain is:

$$\begin{aligned}&(1)\ A + X \to A' \text{ (protection)},\\ &(2)\ A' + B \to C',\\ &(3)\ C' + Y \to C \text{ (deprotection)}.\end{aligned}$$

The trouble with the protection-deprotection steps is that they temporarily make the molecule larger, which means that an algorithm whose aim is to make a molecule smaller will not suggest them.

3. Chemical Graphs

We define a chemical graph as a labelled graph whose edge labels indicate the bond type (either covalent or ionic), and whose vertex labels are either atoms or unpaired electrons together with a charge (Definitions 3.2, 3.5 and 3.6). Chemical graphs are the objects of the reaction category (Definition 5.7) and the disconnection category (Definition 6.5), as well as the targets on which the reaction schemes operate (Definition 5.3 and Theorem 5.5). In order to account for chirality, we add spatial information to chemical graphs, making it an *oriented (pre-)chemical graph* (Definition 3.13) in Subsection 3.1.

Let us fix the following notions needed for Definitions 3.2, 3.5 and 3.6:

- a countable set of *vertex names* **VN**,
- a set of *vertex labels* **At** such that
  (1) **At** is finite,
  (2) **At** contains the special symbol α,

(3) $\mathbf{At} \setminus \{\alpha\}$ has at least two elements,

- a *valence function* $\mathbf{v} : \mathbf{At} \to \mathbb{N}$ such that $\mathbf{v}(\alpha) = 1$,
- the set of *edge labels* $\mathbf{Lab} \coloneqq \{0, 1, 2, 3, 4, \mathtt{i}\}$,
- the functions $\mathtt{cov}, \mathtt{ion} : \mathbf{Lab} \to \mathbb{N}$ defined by
  - if $x \in \{0, 1, 2, 3, 4\}$, then $\mathtt{cov}(x) \coloneqq x$ and $\mathtt{ion}(x) \coloneqq 0$,
  - $\mathtt{cov}(\mathtt{i}) \coloneqq 0$ and $\mathtt{ion}(\mathtt{i}) \coloneqq 1$.

We usually denote the vertex names by lowercase Latin letters or by positive integers. While we only make three formal assumptions about the set of vertex labels, in all the examples we shall assume that **At** contains a symbol for each main-group element of the periodic table: $\{H, C, O, P, \dots\} \subseteq \mathbf{At}$. For this reason, we will also refer to **At** as the *atom labels*. The special symbol α may be thought of as representing an unpaired electron or a free charge. Similarly, we shall assume in the examples that the valence of an element symbol is the number of electrons in its outer electron shell. The integers $\{0, 1, 2, 3, 4\}$ in the set of edge labels stand for covalent bonds, while $\mathtt{i}$ stands for an ionic bond.

Remark 3.1. The reason for choosing such level of generality for the atom labels and their valencies is the ability to model elements which exhibit different valence depending on the context. For instance, one could have separate atom labels for nitrogen whose valence is 5 (all outer shell electrons are shared or take part in a reaction) or 3 (two of the outer shell electrons pair with each other).

Definition 3.2 (Chemically labelled graph)**.** A *chemically labelled graph* is a triple (V, τ, m), where $V \subseteq \mathbf{VN}$ is a finite set of *vertices*, $\tau : V \to \mathbf{At} \times \mathbb{Z}$ is a *vertex labelling function*, and $m : V \times V \to \mathbf{Lab}$ is an *edge labelling function* satisfying $m(v, v) = 0$ and $m(v, w) = m(w, v)$ for all $v, w \in V$.

Thus, a chemically labelled graph is irreflexive (we interpet the edge label 0 as no edge) and symmetric, and each of its vertices is labelled with an element of **At**, together with an integer indicating the charge. Given a chemically labelled graph A, we write (V_A, τ_A, m_A) for its vertex set and the labelling functions. We abbreviate the vertex labelling function followed by the first projection as τ_A^{At}, and similarly we write $\tau_A^{\mathtt{Crg}}$ for composition with the second projection.

Given a chemically labelled graph A and vertex names $u, v \in \mathbf{VN}$ such that $u \in V_A$ but $v \notin V_A \setminus \{u\}$, we denote by $A(u \mapsto v)$ the chemically labelled graph whose vertex set is $(V_A \setminus \{u\}) \cup \{v\}$, and whose vertex and edge labelling functions agree with those of A, treating v as if it were u. Further, we define the following special subsets of vertices:

- *α-vertices*, whose label is the special symbol: $\alpha(A) \coloneqq \tau_A^{-1}(\alpha, \mathbb{Z})$,
- *chemical vertices*, whose label is not α: $\mathtt{Chem}(A) \coloneqq V_A \setminus \alpha(A)$,
- *neutral vertices*, whose charge is zero: $\mathtt{Neu}(A) \coloneqq \tau_A^{-1}(\mathbf{At}, 0)$,
- *charged vertices*, which have a non-zero charge: $\mathtt{Crg}(A) \coloneqq V_A \setminus \mathtt{Neu}(A)$,
- *negative vertices*, which have a negative charge:

$$\mathtt{Crg}^{-}(A) \coloneqq \{v \in V_A : \tau_A^{\mathtt{Crg}}(v) < 0\},$$

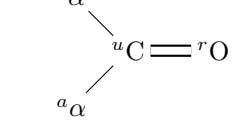
- *positive vertices*, which have a positive charge:

$$\mathtt{Crg}^{+}(A) \coloneqq \{v \in V_A : \tau_A^{\mathtt{Crg}}(v) > 0\}.$$

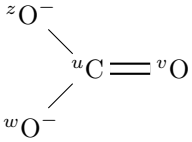
The *net charge* of a subset $U \subseteq V_A$ is the integer $\mathtt{Net}(U) \coloneqq \sum_{v \in U} \tau_A^{\mathtt{Crg}}(v)$.

Example 3.3. We give three examples of chemically labelled graphs: **A**, **B** (carbonate anion) and **C** (sodium cloride). We adopt the following conventions: (1) the vertex label from **At** is drawn at the centre of a vertex, (2) the vertex name is drawn as a superscript on the left (so within a single graph, no left superscript appears twice), (3) a non-zero charge is drawn as a superscript on the right, (4) n-ary covalent bonds are drawn as n parallel lines, and (5) ionic bonds are drawn as dashed lines.

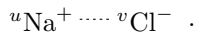
A



B



C



.

Below we give a table with different kinds of vertex subsets for the graphs:

	A	**B**	**C**
α-vertices	$\{a, b\}$	$\varnothing$	$\varnothing$
chemical vertices	$\{r, u\}$	V_B	V_C
neutral vertices	V_A	$\{u, v\}$	$\varnothing$
charged vertices	$\varnothing$	$\{w, z\}$	V_C
negative vertices	$\varnothing$	$\{w, z\}$	$\{v\}$
positive vertices	$\varnothing$	$\varnothing$	$\{u\}$
net charge	0	-2	0

Definition 3.4 (Neighbours)**.** Given a chemically labelled graph A and a vertex $u \in V_A$, we define the sets of *neighbours* $\mathtt{N}_A(u)$, *covalent neighbours* $\mathtt{CN}_A(u)$ and *ionic neighbours* $\mathtt{IN}_A(u)$ of u as follows:

$$\begin{aligned}
\mathtt{N}_A(u) &\coloneqq \{v \in V_A : m_A(u, v) \neq 0\},\\
\mathtt{CN}_A(u) &\coloneqq \{v \in V_A : \mathtt{cov}(m_A(u, v)) \neq 0\},\\
\mathtt{IN}_A(u) &\coloneqq \{v \in V_A : \mathtt{ion}(m_A(u, v)) \neq 0\}.
\end{aligned}$$

Definition 3.5 (Pre-chemical graph)**.** A *pre-chemical graph* $A = (V_A, \tau_A, m_A)$ is a chemically labelled graph satisfying the following additional conditions:

1. for all $v \in \alpha(A)$ and $w \in V_A$ we have
   (a) $\tau_A^{\mathtt{Crg}}(v) \in \{-1, 0, 1\}$,
   (b) $m_A(v, w) \in \{0, 1, \mathtt{i}\}$,
   (c) $\mathtt{N}_A(v)$ has at most one element, and if $w \in \mathtt{N}_A(v)$, then $w \in \mathtt{Chem}(A)$,
2. for all $v \in \mathtt{Chem}(A)$ we have
   (a) either $\mathtt{IN}_A(v) = \{u\}$ for some $u \in \mathtt{Chem}(A)$, or $\mathtt{IN}_A(v) \subseteq \alpha(A) \cap \mathtt{Crg}^{+}(A)$, or $\mathtt{IN}_A(v) \subseteq \alpha(A) \cap \mathtt{Crg}^{-}(A)$,

(b) if $\mathtt{IN}_A(v) \neq \varnothing$, then $v \in \mathtt{Crg}(A)$ and $\tau_A^{\mathtt{Crg}}(v) = -\mathtt{Net}\,(\mathtt{IN}_A(v))$.

Conditions 1a-1c say that a vertex labelled by α is either neutral or has charge ± 1, has at most one neighbour, which is necessarily chemical and to which it is connected either via an ionic or a single covalent bond. Conditions 2a-2b say that an edge with label $\mathtt{i}$ only connects vertices which have opposite charges such that at least one is chemical and the net charges are equal in magnitude.

We say that a pre-chemical graph A is *valence-complete* if for all $v \in V_A$, we have

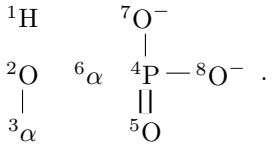
$$\left|\tau_A^{\mathtt{Crg}}(v)\right| + \sum_{u \in V_A} \mathtt{cov}\,(m_A(u,v)) = \mathbf{v}\tau_A^{\mathrm{At}}(v).$$

Definition 3.6. A *chemical graph* is a valence-complete pre-chemical graph A such that $\alpha(A) \cap \mathtt{Crg}^+\,(A) = \varnothing$.

A *synthon* is a chemical graph which is moreover connected. The collection of chemical graphs is, therefore, generated by the disjoint unions of synthons. A *molecular graph* is a chemical graph with no α-vertices. A *molecular entity* is a connected molecular graph.

Example 3.7. The chemically labelled graphs in Example 3.3 are, in fact, chemical graphs with the standard valences of the atoms (i.e. $\mathbf{v}(\mathrm{C}) = 4$, $\mathbf{v}(\mathrm{O}) = 2$ and $\mathbf{v}(\mathrm{H}) = \mathbf{v}(\mathrm{Cl}) = \mathbf{v}(\mathrm{Na}) = 1$). Since the three graphs are connected, they are all synthons. Moreover, **B** and **C** are molecular entities.

Example 3.8. We give an example of a pre-chemical graph which fails to be a chemical graph (the valence of the vertices 1, 2, 4 and 6 is not correct). This graph appears as part of the reaction scheme for glucose phosphorylation in Example 5.2.



3.1. Chirality

An important part of chemical data is stereochemistry, that is, spatial orientation of the molecule: many molecules of interest (like pharmaceuticals) possess chiral enantiomers (i.e. molecules that have the same atoms and connectivity, but are mirror images of each other due to spatial orientation) which have different properties. We therefore wish to incorporate (rudimentary) spatial information into (pre-)chemical graphs. The idea is to record for each triple of atoms whether they are on the same line or not, and similarly, for each quadruple of atoms whether they are in the same plane or not.

While the results of the following sections do not account for orientation and only manipulate the connectivity of a graph, we feel that spatial orientation is such an important aspect of a chemical compound that it has to be accounted for as the immediate next step in this line of work (see Subsection 10.1). We therefore include this subsection, outlining how to incorporate spatial orientation at the level of objects.

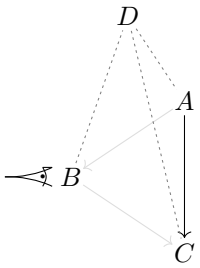


Figure 3: Observer looking at the edge AC from B sees D on their right.

Definition 3.9 (Triangle relation)**.** Let S be a set. We call a ternary relation $\mathcal{P} \subseteq S \times S \times S$ a *triangle relation* if the following hold for all elements A, B and C of S: (1) $ABB \notin \mathcal{P}$, and (2) if $\mathcal{P}(ABC)$ and $\mathfrak{p}(ABC)$ is any permutation of the three elements, then $\mathcal{P}(\mathfrak{p}(ABC))$.

Definition 3.10 (Tetrahedron relation)**.** Let S be a set, and let $\mathcal{P}$ be a fixed triangle relation on S. We call a quaternary relation $\mathcal{T} \subseteq S \times S \times S \times S$ a *tetrahedron relation* if the following hold for all elements A, B, C and D of S: (1) if $\mathcal{T}(ABCD)$, then $\mathcal{P}(ABC)$, and (2) if $\mathcal{T}(ABCD)$ and $\mathfrak{p}(ABCD)$ is any even permutation of the four elements, then $\mathcal{T}(\mathfrak{p}(ABCD))$.

Unpacking the above definitions, a triangle relation is closed under the action of the symmetric group S_3 such that any three elements it relates are pairwise distinct, and a tetrahedron relation is closed under the action of the alternating group A_4 such that if it relates some four elements, then the first three are related by some (fixed) triangle relation (this, inter alia, implies that any related elements are pairwise distinct, and any 3-element subset is related by the fixed triangle relation).

The intuition is that the triangle and tetrahedron relations capture the spatial relations of (not) being on the same line or plane: $\mathcal{P}(ABC)$ stands for A, B and C not being on the same line, that is, determining a triangle; similarly, $\mathcal{T}(ABCD)$ stands for A, B, C and D not being in the same plane, that is, determining a tetrahedron. The tetrahedron is moreover oriented: $\mathcal{T}(ABCD)$ does not, in general, imply $\mathcal{T}(DABC)$. We visualise $\mathcal{T}(ABCD)$ in Figure 3 by placing an "observer" at B who is looking at the edge AC such that A is above C for them. Then D is on the right for this observer. Placing an observer in the same way in a situation where $\mathcal{T}(DABC)$ (which is equivalent to $\mathcal{T}(CBAD)$), they now see D on their left.

Remark 3.11. We chose not to include the orientation of the triangle, which amounts to the choice of S_3 over A_3 in the definition of a triangle relation (Definition 3.9). This is because we assume that our molecules float freely in space (e.g. in a solution), so that there is no two-dimensional orientation.

The following example demonstrates that the triangle and tetrahedron relations indeed capture triangles and tetrahedrons in the Euclidean setting.

Example 3.12. Let us define the triangle relation $\mathcal{P}$ on the 3-dimensional Euclidean space $\mathbb{R}^3$ by letting $\mathcal{P}(abc)$ if and only if $(b-a) \times (c-a) \neq 0$, where $\times$ denotes the vector product. We then have $\mathcal{P}(abc)$ precisely when c does not

lie on the line determined by a and b, that is, when the three points uniquely determine a plane in $\mathbb{R}^3$.

With respect to the above triangle relation, let us define the tetrahedron relation $\mathcal{T}$ by letting $\mathcal{T}(abcd)$ if and only if $\overline{(b-a)(c-a)(d-a)} > 0$, where the bar denotes the scalar triple product. We then have $\mathcal{T}(abcd)$ precisely when the points a, b, c and d are vertices of a non-degenerate (a non-zero volume) tetrahedron in such a way that d lies on that side of the plane determined by a, b and c to which the vector $(b-a) \times (c-a)$ points (see Figure 3).

Definition 3.13 (Oriented chemically labelled graph)**.** An *oriented chemically labelled graph* is a tuple $(A, \mathcal{P}, \mathcal{T})$ where A is a chemically labelled graph, $\mathcal{P}$ is a triangle relation on V_A and $\mathcal{T}$ is a tetrahedron relation on V_A with respect to $\mathcal{P}$.

An *oriented (pre-)chemical graph* is a chemically labelled graph, which is also a (pre-)chemical graph (Definitions 3.5 and 3.6).

Definition 3.14 (Preservation and reflection of orientation)**.** Let $(M, \mathcal{P}_M, \mathcal{T}_M)$ and $(N, \mathcal{P}_N, \mathcal{T}_N)$ be oriented chemically labelled graphs, and let $f : M \to N$ be a labelled graph isomorphism. We say that f *preserves orientation* (or is *orientation-preserving*) if for all vertices A, B, C and D of M we have:

(1) $\mathcal{P}_M(ABC)$ if and only if $\mathcal{P}_N(fA, fB, fC)$, and

(2) $\mathcal{T}_M(ABCD)$ if and only if $\mathcal{T}_N(fA, fB, fC, fD)$.

Similarly, we say that f *reflects orientation* (or is *orientation-reflecting*) if for all vertices A, B, C and D of M we have:

(1) $\mathcal{P}_M(ABC)$ if and only if $\mathcal{P}_N(fA, fB, fC)$, and

(2) $\mathcal{T}_M(ABCD)$ if and only if $\mathcal{T}_N(fD, fA, fB, fC)$.

Note that an orientation-reflecting isomorphism differs from an orientation-preserving one only in that preservation requires the tetrahedron relations to be the same (up to an even permutation), while reflection requires them to be the same up to an odd permutation.

Definition 3.15 (Chirality)**.** We say that two oriented chemically labelled graphs are *chiral* if there exists an orientation-reflecting isomorphism, but no orientation-preserving isomorphism between them.

Example 3.16. Consider 2-butanol, whose molecular structure we draw in two different ways at the top of Figure 4. Here we adopt the usual chemical convention for drawing spatial structure: a dashed wedge indicates that the bond points "into the page", and a solid wedge indicates that the bond points "out of the page". The spatial structure is formalised by defining the tetrahedron relation for the graph on the left-hand side as the closure under the action of A_4 of $\mathcal{T}(1234)$, and for the one on the right-hand side as (the closure of) $\mathcal{T}(4123)$. In both cases, the triangle relation is dictated by the tetrahedron relation, so that any three-element subset of $\{1, 2, 3, 4\}$ is in the triangle relation. Now the identity map (on labelled graphs) reflects orientation. It is furthermore not hard to see that every isomorphism restricts to the identity on the vertices labelled

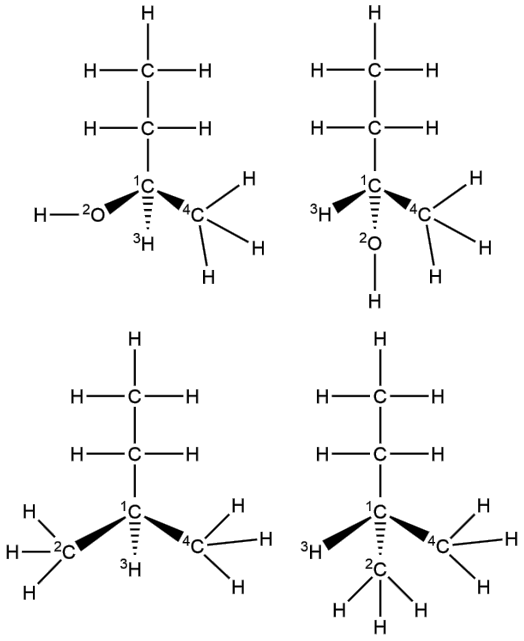
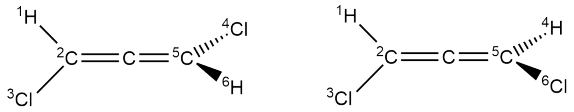


Figure 4: Top: two configurations of 2-butanol. Bottom: two configurations of isopentane.

with superscripts, so that there is no orientation-preserving isomorphism. Thus the two molecules are chiral according to Definition 3.15.

By slightly modifying the structures, we obtain two configurations of isopentane, drawn at the bottom of Figure 4. However, in this case we can find an orientation-preserving isomorphism (namely the one that swaps vertices 2 and 4), so that the molecules are not chiral.

Example 3.17. Example 3.16 with 2-butanol demonstrated how to capture central chirality using Definition 3.15. Now consider 1,3-dichloroallene as an example of axial chirality. We draw two versions, as before:



The tetrahedron relation is generated by $\mathcal{T}(1234)$ and $\mathcal{T}(6123)$ for both molecules (note, however, that the vertices 4 and 6 have different labels). Now the isomorphism which swaps vertices 4 and 6 and is identity on all other vertices is orientation-reflecting, but not orientation-preserving. The only other isomorphism is $\{1 \mapsto 4, 2 \mapsto 5, 3 \mapsto 6, 4 \mapsto 3, 5 \mapsto 2, 6 \mapsto 1\}$, which does not preserve orientation. Thus the two molecules are indeed chiral.

4. Pre-Chemical Graphs Form an Adhesive Category

In this section we equip pre-chemical graphs with a notion of morphisms, making them into a category. We identify two classes of morphisms, *vertex*

embeddings (Definition 4.4) and *matchings* (Definition 4.7), which are used for double pushout rewriting in the next section. As a mathematical prerequisite for double pushout graph rewriting, we prove in Subsection 4.1 that vertex embeddings and matchings give the category of pre-chemical graphs an adhesive structure (Theorem 4.24).

A morphism of pre-chemical graphs is a function which, intuitively, preserves any resources and structure (matter, charge and bonds) present in the domain. While the main notion of a chemically meaningful transformation is that of a reaction (Definition 5.7), it is not a natural notion of a graph morphism, as it only captures a subclass of bijective maps. The morphisms are needed to capture reaction schemes (Definition 5.1) as well as their instances (Definition 5.3), which we show to be in one-to-one correspondence with reactions (with some mild assumptions on the subsets changed by the reaction) in Proposition 5.6.

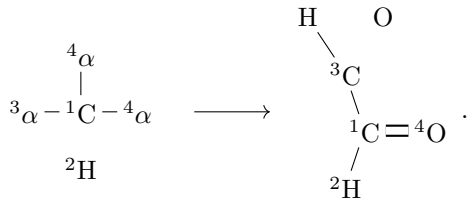
Definition 4.1 (Morphism of pre-chemical graphs)**.** A *morphism* of pre-chemical graphs $f : A \to B$ is a function $f : V_A \to V_B$ such that its restriction to the chemical vertices $f|_{\mathtt{Chem}(A)}$ is injective, the images $f(\mathtt{Chem}(A))$ and $f(\alpha(A))$ are disjoint, and for all $v, u \in V_A$ and $w, z \in V_B$ we have

1. if $v \in \mathtt{Chem}(A)$, then $\tau_B^{\mathrm{At}}(fv) = \tau_A^{\mathrm{At}}(v)$,
2. if $v \in \mathtt{Crg}^+(A)$, then $f(v) \in \mathtt{Crg}^+(B)$; and if $v \in \mathtt{Crg}^-(A)$, then $f(v) \in \mathtt{Crg}^-(B)$,
3. if $\mathtt{Net}\left(f^{-1}(w)\right) \neq 0$, then $\mathtt{Net}\left(f^{-1}(w)\right) = \tau_B^{\mathtt{Crg}}(w)$,
4. if $m_A(v, u) = \mathtt{i}$, then $m_B(fv, fu) = \mathtt{i}$,
5. if $v, u \in \mathtt{Chem}(A)$ and $m_A(v, u) \neq 0$, then $m_B(fv, fu) = m_A(v, u)$,
6. if $w \in f(\alpha(A))$ and $z = f(b)$ for some $b \in \mathtt{Chem}(A)$ such that

$$k \coloneqq \sum_{a \in f^{-1}(w)} \mathtt{cov}(m_A(a, b)) \neq 0,$$

then $k = \mathtt{cov}(m_B(w, z))$.

Example 4.2. The idea of a morphism is that it preserves all the atoms, bonds and charges present in the domain, potentially more being present in the codomain. We give an example below, where we use superscripts to indicate the underlying function: each vertex in the domain is mapped to the vertex in the codomain with the same superscript:



Let us denote by **PChem** the category of pre-chemical graphs and their morphisms. Note that **PChem** has a symmetric monoidal structure given by the disjoint union of chemical graphs.

Proposition 4.3. *If $f : A \to B$ is a morphism of pre-chemical graphs, then for all $a, v \in V_A$, we have*

(1) $\mathtt{cov}\,(m_A(a,v)) \leq \mathtt{cov}\,(m_B(fa,fv))$,

(2) $\sum_{u\in V_A} \mathtt{cov}\,(m_A(a,u)) \leq \sum_{u\in V_B} \mathtt{cov}\,(m_B(fa,u))$,

(3) $f\,(\mathtt{CN}_A(a)) \subseteq \mathtt{CN}_B(fa)$.

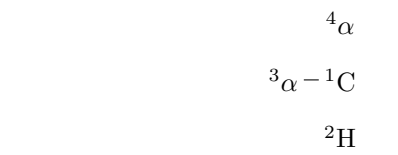
Proof. If $\mathtt{cov}\,(m_A(a,v)) = 0$, then (1) is immediate. Hence let $v \in \mathtt{CN}_A(a)$. By condition 1c of being a pre-chemical graph (Definition 3.5), at least one of a and v is chemical. If both are chemical, then condition 5 of Definition 4.1 yields $m_B(fa,fv) = m_A(a,v)$, so that (1) holds. Hence suppose that exactly one is an α-vertex: without loss of generality, suppose that $a \in \alpha(A)$ and $v \in \mathtt{Chem}(A)$. Condition 6 Definition 4.1 then yields

$$m_B(fa,fv) = \sum_{u\in f^{-1}f(a)} m_A(u,v) \geq m_A(a,v),$$

as is required for (1). Items (2) and (3) are easy consequences of ((1)). □

Definition 4.4 (Vertex embedding)**.** We say that a morphism $f : A \to B$ in **PChem** is a *vertex embedding* if it is injective, bijective on chemical vertices, and for all $u \in V_A$ we have $\tau_B^{\mathrm{At}}(fu) = \tau_A^{\mathrm{At}}(u)$.

Example 4.5. A vertex embedding is an injective morphism that preserves all the atom labels (including α) such that all the chemical vertices of the codomain are in its image. In other words, it can only add new charges, edges or α-vertices to the domain graph. We give an example below:

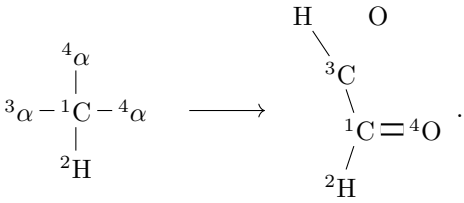


We denote the class of vertex embeddings by $\mathcal{E}$, and will often refer to the elements of $\mathcal{E}$ simply as *embeddings*.

Definition 4.6 (Ion-closed subset)**.** Let A be a pre-chemical graph. A subset $U \subseteq V_A$ is *ion-closed* if for all vertices $u, v \in V_A$, if $u \in U$ and $m_A(u,v) = \mathtt{i}$, then also $v \in U$.

Definition 4.7 (Matching)**.** A *matching* is a morphism $f : A \to C$ in **PChem** such that the conditions (3), (5) and (6) of being a morphism (Definition 4.1) hold without the exception for the zero charge or bond case, and the image $f(V_A)$ is ion-closed.

Example 4.8. A matching is a morphism with the further restrictions that all charges are preserved (including the zero charge), no new bonds can be added between existing vertices. We thus think of a matching as identifying the domain as a substructure of the codomain. We slightly modify the morphism in Example 4.2 to obtain a matching:



Let us denote the class of matchings by $\mathcal{M}$. We wish to characterise the images of matchings between valence-complete pre-chemical graphs, which we achieve in Proposition 4.16. To this end, we need to define the *valence completion* (Definition 4.10) and *charge decomposition* (Definition 4.12) of a subset of a valence-complete pre-chemcial graph. We begin with the following observation.

Lemma 4.9. *If $m : A \to C$ is a matching whose domain and codomain are valence-complete, then for every $u \in \mathtt{Chem}(A)$ we have $m\left(\mathtt{CN}_A(u)\right) = \mathtt{CN}_C(mu)$.*

Proof. The inclusion $m\left(\mathtt{CN}(u)\right) \subseteq \mathtt{CN}(mu)$ is part (3) of Proposition 4.3. Since $\tau_A(u) = \tau_C(mu)$, we have

$$\sum_{w \in \mathtt{CN}(u)} m_A(u,w) = \sum_{w \in \mathtt{CN}(mu)} m_C(mu,w).$$

If $w \in \mathtt{CN}(u)$ is chemical, then $m_C(mu,mw) = m_A(u,w)$, while if it is an α-vertex, then $m_C(mu,mw) = \sum_{z \in m^{-1}m(w)} m_A(u,z)$. It follows that

$$\begin{aligned}
\sum_{w \in \mathtt{CN}(u)} m_A(u,w) &= \sum_{w \in \mathtt{CN}(u) \cap \mathtt{Chem}(A)} m_A(u,w) + \sum_{w \in \mathtt{CN}(u) \cap \alpha(A)} m_A(u,w) \\
&= \sum_{w \in \mathtt{CN}(u) \cap \mathtt{Chem}(A)} m_C(mu,mw) \\
&+ \sum_{w \in m(\mathtt{CN}(u) \cap \alpha(A))} \sum_{z \in m^{-1}(w)} m_A(u,z) \\
&= \sum_{w \in m(\mathtt{CN}(u) \cap \mathtt{Chem}(A))} m_C(mu,w) + \sum_{w \in m(\mathtt{CN}(u) \cap \alpha(A))} m_C(mu,w) \\
&= \sum_{w \in m(\mathtt{CN}(u))} m_C(mu,w).
\end{aligned}$$

Combining this with the first equality, we get that

$$\sum_{w \in \mathtt{CN}(mu)} m_C(mu,w) = \sum_{w \in m(\mathtt{CN}(u))} m_C(mu,w).$$

Since $m\left(\mathtt{CN}(u)\right) \subseteq \mathtt{CN}(mu)$, it follows that the sets are, in fact, equal. □

Definition 4.10 (Valence completion)**.** Let A be a valence-complete pre-chemical graph and let $U \subseteq \mathtt{Chem}(A)$. We define the sets of formal symbols called the *indexed covalent neighbours*, and the *indexed ionic neighbours* as follows:

$$\begin{aligned}
\mathcal{CN}(U) &\coloneqq \left\{ v_j^u : u \in U, v \in \mathtt{CN}_A(u) \cap (V_A \setminus U) \text{ and } j = 1, \ldots, m_A(u,v) \right\}, \\
\mathcal{IN}(U) &\coloneqq \left\{ v_j^{u,\mathtt{i}} : u \in U, v \in \mathtt{IN}_A(u) \cap (V_A \setminus U) \text{ and } j = 1, \ldots, \left| \tau_A^{\mathtt{Crg}}(v) \right| \right\}.
\end{aligned}$$

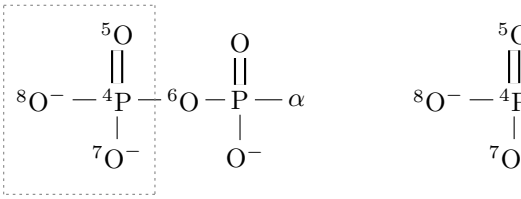
The *valence completion* of U is the pre-chemical graph[2]

$$U^{\alpha} \coloneqq (U \cup \mathcal{CN}(U) \cup \mathcal{IN}(U), \tau_{\alpha}, m_{\alpha})$$

defined by the following labelling functions: for $u, w \in U$ and $z \in \mathcal{CN}(U) \cup \mathcal{IN}(U)$, let $\tau_{\alpha}(u) \coloneqq \tau_A(u)$, $m_{\alpha}(u,w) \coloneqq m_A(u,w)$ and $\tau_{\alpha}^{\mathrm{At}}(z) \coloneqq \alpha$, and for the sets defined above we let

$$\begin{aligned}
\tau_{\alpha}^{\mathtt{Crg}}(v_j^u) &\coloneqq 0, \\
\tau_{\alpha}^{\mathtt{Crg}}(v_j^{u,\mathtt{i}}) &\coloneqq \begin{cases} 1 \text{ if } \tau_A^{\mathtt{Crg}}(v) > 0, \\ -1 \text{ if } \tau_A^{\mathtt{Crg}}(v) < 0, \end{cases} \\
m_{\alpha}(u, v_j^u) &\coloneqq 1, \\
m_{\alpha}(u, v_j^{u,\mathtt{i}}) &\coloneqq \mathtt{i}.
\end{aligned}$$

Example 4.11. Consider the chemical graph below left. The valence completion of the dashed subset is given on the right. Note that there is a matching from the valence completion into the original graph given by the identity on the vertices.



Definition 4.12 (Charge decomposition)**.** Let A be a valence complete pre-chemical graph and let $B \subseteq \mathtt{Crg}(A)$ be a subset of charged vertices. We define the following sets of formal symbols: *indexed positive vertices*, and *indexed negative vertices* as follows:

$$\begin{aligned}
\mathcal{PV}(B) &\coloneqq \left\{ b_j^+ : b \in B \cap \mathtt{Crg}^+(A) \text{ and } j = 1, \ldots, \left| \tau_A^{\mathtt{Crg}}(b) \right| \right\}, \\
\mathcal{NV}(B) &\coloneqq \left\{ b_j^- : b \in B \cap \mathtt{Crg}^-(A) \text{ and } j = 1, \ldots, \left| \tau_A^{\mathtt{Crg}}(b) \right| \right\}.
\end{aligned}$$

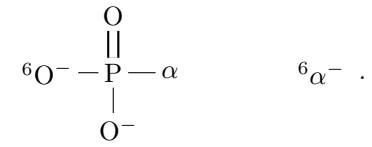
The *charge decomposition* of U is the pre-chemical graph

$$B^{\mathtt{Crg}} \coloneqq \left(\mathcal{PV}(B) \cup \mathcal{NV}(B), \tau_{\mathtt{Crg}}, m_{\mathtt{Crg}}\right)$$

defined by $\tau_{\mathtt{Crg}}\left(b_j^+\right) \coloneqq (\alpha, 1)$, $\tau_{\mathtt{Crg}}\left(b_j^-\right) \coloneqq (\alpha, -1)$, and $m_{\mathtt{Crg}}$ is the constantly zero function.

Example 4.13. Consider the chemical graph below left. The charge decomposition of the oxygen vertex with vertex name 6 is given by the single negatively charged α-vertex drawn on the right. As for the charge decomposition, note that the identity mapping is a matching.

[2]We assume a unique choice of a vertex name for every element in $\mathcal{CN}(U) \cup \mathcal{IN}(U)$, disjoint from the vertex names of V_A. For legibility, we omit this technical detail.



We now define the notion of a *matchable subset*, which lists the conditions under which a subset of a valence-complete chemical graph comes from a matching.

Definition 4.14 (Matchable subset)**.** Let A be a valence-complete pre-chemical graph. We say that a subset $U \subseteq V_A$ is *matchable* if it is ion-closed, and for every $u \in U$, either $u \in \mathtt{Chem}(U)$ and $\mathtt{N}(u) \subseteq U$, or $u \in \mathtt{Crg}(A)$, or there is a $v \in \mathtt{Chem}(U) \cap \mathtt{N}(u)$ with $\mathtt{N}(v) \subseteq U$.

Thus a subset is matchable if all boundary vertices and α-vertices are either charged or connected to at least one interior chemical vertex. We first observe that being matchable is a necessary condition for a subset to be an image of a matching.

Lemma 4.15. *First, $m(A)$ is ion-closed by definition. Let $m : A \to B$ be a matching such that A and B are valence-complete. Then $m(A)$ is a matchable subset of B.*

Proof. If $u \in \mathtt{Chem}(A)$, then $\mathtt{N}_B(mu) = m(\mathtt{N}_A(u)) \subseteq m(A)$. If $u \in \alpha(A) \cap \mathtt{Crg}(A)$, then $m(u) \in \mathtt{Crg}(B)$. If $u \in \alpha(A) \cap \mathtt{Neu}(A)$, then there is a $v \in \mathtt{Chem}(A)$ with $m_A(u, v) = 1$, so that $m(v) \in \mathtt{Chem}(m(A)) \cap \mathtt{N}_B(mu)$. □

Proposition 4.16. *Let A be a valence-complete pre-chemical graph. A subset $S \subseteq V_A$ is matchable if and only if there is a matching $m : C \to A$ with a valence-complete domain such that $m(C) = S$.*

Proof. The 'if' direction is Lemma 4.15. Hence suppose S is matchable. We define the following sets:

$$U \coloneqq \{s \in \mathtt{Chem}(S) : \mathtt{N}(s) \subseteq S\},$$
$$B \coloneqq \mathtt{Crg}(S \setminus (U \cup \mathtt{IN}(U))).$$

It follows that $U \cup \mathtt{N}(U) \cup B = S$, so that the image of the matching $U^\alpha + B^{\mathtt{Crg}} \to A$ is precisely S. □

4.1. Adhesivity

Adhesive categories were introduced by Lack and Sobociński [32, 33] as a categorical setting where pushouts along monomorphisms are well-behaved. The main motivation is to provide an abstract mathematical framework for double pushout graph rewriting. Adhesive categories have been generalised to $\mathcal{M}$-adhesive categories by Ehrig, Golas and Hermann [34], allowing for good behaviour of pushouts along a specified subclass of monomorphisms. A further generalisation restricts the class of morphisms that have pushouts (still along a restricted class of monomorphisms), resulting in $(\mathcal{M}, \mathcal{N})$-adhesive categories of Habel and Plump [35]. Here we prove that the category of pre-chemical graphs **PChem** is $(\mathcal{E}, \mathcal{M})$-adhesive (Theorem 4.24), where $\mathcal{E}$ are the vertex embeddings, and $\mathcal{M}$ are the matchings. This result enables double pushout rewriting

in **PChem**, which we will use in the next section to define reaction schemes and their instances.

We begin by stating the definition of an $(\mathcal{M},\mathcal{N})$-adhesive category. For the sake of brevity, we do so without a detailed discussion of the terms appearing in the definition. We refer the reader to Habel and Plump [35], Castelnovo, Gadducci and Miculan [36], and Castelnovo and Miculan [37] for the details.

Definition 4.17 ($(\mathcal{M},\mathcal{N})$-adhesive category. Definition 1 in [35])**.** Let $\mathcal{C}$ be a category, $\mathcal{M}$ a class of monomorphisms and $\mathcal{N}$ a class of morphisms in $\mathcal{C}$. We say that $\mathcal{C}$ is $(\mathcal{M},\mathcal{N})$-*adhesive* if the following properties hold:

1. $\mathcal{M}$ and $\mathcal{N}$ contain all isomorphisms and are closed under composition and decomposition. Moreover, $\mathcal{N}$ is closed under $\mathcal{M}$-decomposition: if $g \circ f \in \mathcal{N}$ and $g \in \mathcal{M}$, then $f \in \mathcal{N}$.

2. $(\mathcal{M},\mathcal{N})$-pushouts and pullbacks along $\mathcal{M}$-morphisms exist in $\mathcal{C}$. Also, $\mathcal{M}$ and $\mathcal{N}$ are stable under $(\mathcal{M},\mathcal{N})$-pushouts and $\mathcal{M}$-pullbacks.

3. $(\mathcal{M},\mathcal{N})$-pushouts are $(\mathcal{M},\mathcal{N})$-van Kampen squares.

Let us denote by $\mathcal{P}_{fin}(\mathbf{VN})$ the category whose objects are the finite subsets of **VN**, and whose morphisms are functions (so the category is equivalent to the usual category of finite sets). Let $U : \mathbf{PChem} \to \mathcal{P}_{fin}(\mathbf{VN})$ denote the evident forgetful functor. We also write $U : \mathbf{PChem} \to \mathbf{Set}$ for the forgetful functor into the category of sets and functions. Now define a functor $F : \mathcal{P}_{fin}(\mathbf{VN}) \to \mathbf{PChem}$ by letting $F(V) = (V, \tau^\alpha, m^0)$, where τ^α and m^0 are constant functions sending every element to $(\alpha, 0)$ and 0, respectively. A function $f : V \to W$ is mapped to itself: note that since $F(V)$ does not have any chemical or charged vertices, all the conditions of Definition 4.1 trivialise, so that any function with $F(V)$ as the domain is a morphism of chemical graphs.

Proposition 4.18. *The functors*

$$F : \mathcal{P}_{fin}(\mathbf{VN}) \rightleftarrows \mathbf{PChem} : U$$

are adjoint with $F \dashv U$*.*

Proof. The unit $\eta_V : V \to UF(V)$ is given by the identity, as $UF(V) = V$. The counit $\varepsilon_A : FU(A) \to A$ is given by the identity function: the conditions of being a morphism are automatically verified. It is then straightforward to verify the triangle identities. □

Corollary 4.19. *The forgetful functor* $U : \mathbf{PChem} \to \mathbf{Set}$ *preserves all finite limits that exist in* **PChem***.*

Proposition 4.20. *The pullbacks along vertex embeddings exist in* **PChem***. Moreover, the vertex embeddings are stable under pullbacks, and matchings are stable under* $\mathcal{E}$*-pullbacks.*

Proof. Consider the cospan $A \xrightarrow{f} B \xleftarrow{e} C$ where $e \in \mathcal{E}$. Define the chemical graph Z as follows:

- $V_Z \coloneqq \{a \in V_A : f(a) \in e(V_C)\}$
- for all $a \in V_Z$, let $\tau_Z^{\mathrm{At}}(a) \coloneqq \tau_A^{\mathrm{At}}(a)$,

- for $a \in \mathtt{Chem}(Z)$, if $\tau_A^{\mathtt{Crg}}(a) = \tau_C^{\mathtt{Crg}}(e^{-1}f(a)) =: n$, then let $\tau_Z^{\mathtt{Crg}}(a) \coloneqq n$, otherwise let $\tau_Z^{\mathtt{Crg}}(a) \coloneqq 0$,
- for $a \in \alpha(Z)$, if both $a \in \mathtt{Crg}(A)$ and $e^{-1}f(a) \in \mathtt{Crg}(C)$, then let $\tau_Z^{\mathtt{Crg}}(a) \coloneqq \tau_A^{\mathtt{Crg}}(a)$, otherwise let $\tau_Z^{\mathtt{Crg}}(a) \coloneqq 0$,
- for $a, q \in \mathtt{Chem}(Z)$, if $m_A(a,q) = m_C(e^{-1}f(a), e^{-1}f(q)) =: k$, let $m_Z(a,q) \coloneqq k$, otherwise let $m_Z(a,q) \coloneqq 0$,
- for $a \in \alpha(Z)$ and $q \in \mathtt{Chem}(Z)$, if both $a \in \mathtt{CN}_A(q)$ and $e^{-1}f(a) \in \mathtt{CN}_C(e^{-1}f(a))$, then let $m_Z(a,q) \coloneqq 1$, otherwise let $m_Z(a,q) \coloneqq 0$.

Define the map $e^* : Z \to A$ as the identity on vertices, and the map $f^* : Z \to C$ by the action of $e^{-1}f$ on vertices. Then the resulting square commutes by construction, and the universal property of a pullback is readily verified. Evidently, $e^* \in \mathcal{E}$.

If $f \in \mathcal{M}$, then $f^*(V_Z)$ is ion-closed since $f(V_A)$ is, and f^* preserves all charges and bonds since f does and by construction of Z, so that $f^* \in \mathcal{M}$. □

Lemma 4.21. *The forgetful functor $U : \mathbf{PChem} \to \mathbf{Set}$ preserves and reflects $\mathcal{E}$-pullbacks. In more detail, the commutative square on the left with $e, e' \in \mathcal{E}$ is a pullback in* **PChem** *if and only if its image on the right is a pullback in* **Set***:*

$$\begin{array}{ccc} A & \xrightarrow{f'} & B \\ {\scriptstyle e'}\downarrow & & \downarrow{\scriptstyle e} \\ C & \xrightarrow{f} & D \end{array} \qquad \begin{array}{ccc} U(A) & \xrightarrow{Uf'} & U(B) \\ {\scriptstyle Ue'}\downarrow & & \downarrow{\scriptstyle Ue} \\ U(C) & \xrightarrow{Uf} & U(D) \end{array} .$$

Proof. Preservation is a special case of Corollary 4.19. For reflection, suppose that the square on the right is a pullback in **Set**. This means there is a unique isomorphism from $u : U(A) \to U(Z)$ to the pullback of $C \xrightarrow{f} D \xleftarrow{e} B$ as constructed in the proof of Proposition 4.20 such that $Uf^* \circ u = Uf'$ and $Ue^* \circ u = Ue'$. Using the properties of the pullback and the fact that e' is an embedding, one then checks that u is an isomorphism in **PChem**. □

Proposition 4.22. PChem *has $(\mathcal{E}, \mathcal{M})$-pushouts, and the classes $\mathcal{E}$ and $\mathcal{M}$ are stable under $(\mathcal{E}, \mathcal{M})$-pushouts.*

Proof. Consider the span $B \xleftarrow{m} A \xrightarrow{e} C$ where $m \in \mathcal{M}$ and $e \in \mathcal{E}$. Define the chemical graph Y whose vertex set is that of B together with the complement of the image of e:

$$V_Y \coloneqq V_B \cup (V_C \setminus e(V_A)).$$

Note that $V_C \setminus e(V_A)$ only contains α-vertices. The labelling functions are defined as follows for all $b, p \in V_B$ and $c \in V_C \setminus e(V_A)$:

- $\tau_Y^{\mathrm{At}}(b) \coloneqq \tau_B^{\mathrm{At}}(b)$ and $\tau_Y(c) \coloneqq \tau_C(c)$,
- if $b \in m(A)$, then $\tau_Y^{\mathtt{Crg}}(b) \coloneqq \sum_{d \in em^{-1}(b)} \tau_C^{\mathtt{Crg}}(d)$, and $\tau_Y^{\mathtt{Crg}}(b) \coloneqq \tau_B^{\mathtt{Crg}}(b)$ otherwise,

- if $b, p \in m(\mathtt{Chem}(A))$, then $m_Y(b,p) \coloneqq m_C(em^{-1}(b), em^{-1}(p))$; and if $b \in m(\mathtt{Chem}(A))$ and $p \in m(\alpha(A))$, then

$$m_Y(b,p) \coloneqq \sum_{d \in em^{-1}(p)} m_C(em^{-1}(b), d),$$

  and $m_Y(b,p) = m_B(b,p)$ otherwise,

- if $b \in m(\mathtt{Chem}(A))$, then $m_Y(b,c) \coloneqq m_C(em^{-1}(b), c)$ and $m_Y(b,c) \coloneqq 0$ otherwise.

The map $e^* : B \to Y$ is defined as inclusion on vertices, and the map $m^* : C \to Y$ as me^{-1} on the image of e, and as inclusion otherwise. Then, by construction, the resulting square commutes and we have $e^* \in \mathcal{E}$ and $m^* \in \mathcal{M}$, and the universal property of a pushout is readily verified. □

Lemma 4.23. *The forgetful functor $U : \mathbf{PChem} \to \mathbf{Set}$ preserves and reflects $(\mathcal{E}, \mathcal{M})$-pushouts. In more detail, the commutative square on the left with $e, e' \in \mathcal{E}$ and $m, m' \in \mathcal{M}$ is a pushout in* **PChem** *if and only if its image on the right is a pushout in* **Set***:*

$$\begin{array}{ccc} A & \xrightarrow{m} & B \\ {\scriptstyle e}\downarrow & & \downarrow{\scriptstyle e'} \\ C & \xrightarrow{m'} & D \end{array} \qquad \begin{array}{ccc} U(A) & \xrightarrow{Um} & U(B) \\ {\scriptstyle Ue}\downarrow & & \downarrow{\scriptstyle Ue'} \\ U(C) & \xrightarrow{Um'} & U(D) \end{array}.$$

Proof. For preservation, observe that $U(Y)$ is the pushout of $U(B) \xleftarrow{Um} U(A) \xrightarrow{Ue} U(C)$, where Y is the pushout of $B \xleftarrow{m} A \xrightarrow{e} C$ as constructed in the proof of Proposition 4.23. For reflection, suppose that the square on the right is a pushout in **Set**. This means there is a unique isomorphism from $u : U(Y) \to U(D)$ such that $u \circ Um^* = Um'$ and $u \circ Ue^* = Ue'$. Using the properties of the pushout and the facts that e' is an embedding and m' is a matching, one then checks that u is an isomorphism in **PChem**. □

Theorem 4.24. *The category* **PChem** *is $(\mathcal{E}, \mathcal{M})$-adhesive.*

Proof. It is straightforward to see that both $\mathcal{E}$ and $\mathcal{M}$ contain all isomorphisms, and are closed under composition and decomposition.

To see that $\mathcal{M}$ is closed under $\mathcal{E}$-decomposition, suppose that $g \in \mathcal{E}$ with type $g : B \to C$ and for some composable morphism $f : A \to B$ we have $gf \in \mathcal{M}$. The fact that $f(V_A)$ is ion-closed follows from ion-closedness of $gf(V_A)$ and from injectivity of g. Since gf preserves the labels of chemical vertices (and of edges between chemical vertices), and the only charge (edge label) that can be mapped by g to zero is zero, we conclude that f must preserve the labels of chemical vertices and of the edges between them. For the last two conditions of being a matching, let $w \in f(\alpha(A))$ and $b \in \mathtt{Chem}(A)$. Since gf is a matching and g is an embedding, we have

$$\tau_C^{\mathtt{Crg}}(gw) = \sum_{a \in f^{-1}(w)} \tau_A^{\mathtt{Crg}}(a) =: n.$$

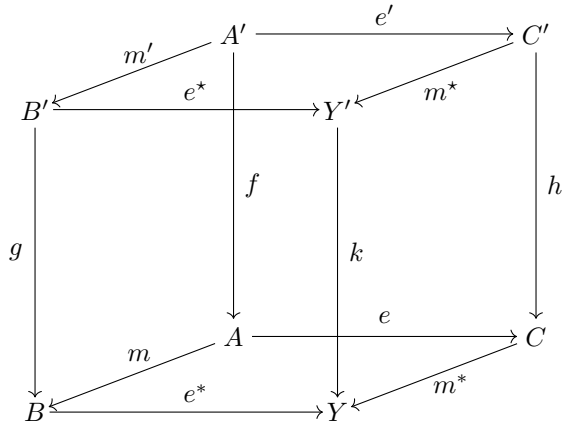
If $n = 0$, then $\tau_C^{\mathtt{Crg}}(gw) = \tau_B^{\mathtt{Crg}}(w)$, so that the desired equality holds. If $n < 0$, then $\tau_B^{\mathtt{Crg}}(w) \leq n$ as f is a morphism, but also $n \leq \tau_B^{\mathtt{Crg}}(w)$ as g is a morphism, so that $\tau_B^{\mathtt{Crg}}(w) = n$ and the desired equality holds. The case for $n < 0$ is symmetric.

For the last condition, we again use the facts that gf is a matching and g is an embedding to obtain

$$m_C\left(g(w), gf(b)\right) = \sum_{a \in f^{-1}(w)} m_A(a, b) =: d.$$

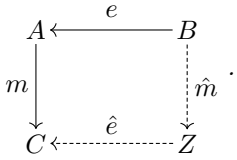
If either $d = 0$ or $w \in \mathtt{Chem}(B)$, then $m_C\left(g(w), gf(b)\right) = m_B\left(w, f(b)\right)$, so that the desired equality holds. Hence suppose that $d > 0$ and $w \in \alpha(B)$. It follows that there is exactly one $a \in f^{-1}(w)$ such that $m_A(a, b) = 1$, and $d = m_B\left(w, f(b)\right) = 1$, so that the desired equality holds. Thus indeed $f \in \mathcal{M}$.

Existence of $\mathcal{E}$-pullbacks, as well as stability of $\mathcal{E}$ and $\mathcal{M}$ under ($\mathcal{E}$-)pullbacks was shown in Proposition 4.20. Existence of $(\mathcal{E}, \mathcal{M})$-pushouts and stability of $\mathcal{E}$ and $\mathcal{M}$ under such pushouts is Proposition 4.22. Thus it remains to show that $(\mathcal{E}, \mathcal{M})$-pushouts are van Kampen squares. Consider the commutative cube



whose bottom face is a pushout of an embedding e and a matching m, whose back faces are both pullbacks, and whose all vertical morphisms are embeddings. We thus have that $m^*, m' \in \mathcal{M}$ and $e^*, e' \in \mathcal{E}$. We have to show that the top face is a pushout if and only if the front faces are pullbacks. By Lemmas 4.21 and 4.23, the top face is a pushout in **PChem** if and only if it is a pushout in **Set**, and the front faces are pullbacks in **PChem** if and only if they are pullbacks in **Set**. Lemma 4.23, the bottom face is a pushout in **Set**. Since in **Set** pushouts along monomorphisms are van Kampen squares [33], we indeed have that the top face is a pushout in **Set** if and only if the front faces are pullbacks in **Set**, thus completing the proof. □

Corollary 4.25 (Pushout complements)**.** *The solid diagram below, where $e \in \mathcal{E}$ and $m \in \mathcal{M}$, can be uniquely completed to a pushout square, with $\hat{e} \in \mathcal{E}$ and $\hat{m} \in \mathcal{M}$:*



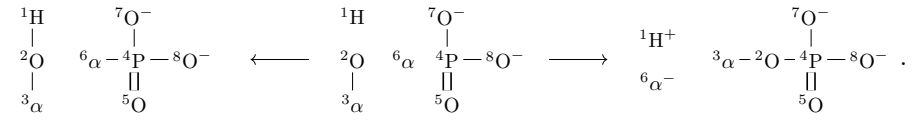
Proof. The pre-chemical graph Z is defined to have the vertex set $V_C \setminus m(V_A \setminus e(V_B))$. The atom labels are inherited from C, and likewise for the charge and edge labels on $V_C \setminus m(V_A)$, while on $m(V_A)$ the charge and edge labels are those of B. Uniqueness is a consequence of $(\mathcal{E}, \mathcal{M})$-adhesivity [35]. □

5. Chemical Reactions as Double Pushout Rewriting

In this section, we first give a formalisation of chemical reactions using double pushout rewriting – our first perspective on chemical processes. Our approach is very similar, and inspired by, that of Andersen, Flamm, Merkle and Stadler [38], with some important differences, such as having more strict requirements on the graphs representing molecular entities, and allowing for the free and unpaired electrons, represented by the symbol α. After this, we characterise all the possible graph transformations resulting from such double pushout rewriting as certain partial bijections (Proposition 5.6). This gives rise to our second perspective on chemical processes – the category of reactions (Definition 5.7).

Definition 5.1 (Reaction scheme)**.** A *reaction scheme* is a span $A \xleftarrow{f} K \xrightarrow{g} B$ in **PChem** such that A and B are valence-complete and have the same net charge, f and g are vertex embeddings, and the span is terminal in the subcategory of spans with boundaries A and B and legs in $\mathcal{E}$.

Example 5.2. The rule shown below appears in the reaction describing glucose phosphorylation. It is a reaction scheme in the sense of Definition 5.1:

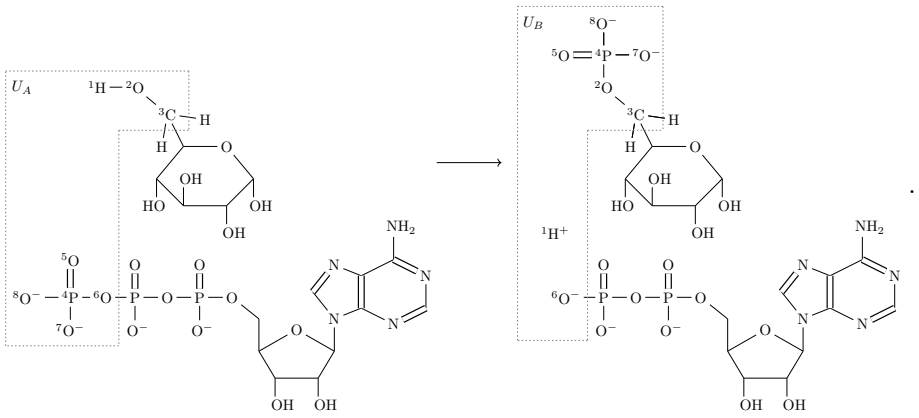


Definition 5.3 (Reaction instance)**.** A *reaction instance* is a double pushout diagram

$$\begin{array}{ccccc} A & \xleftarrow{\quad f \quad} & K & \xrightarrow{\quad g \quad} & B \\ {\scriptstyle m}\downarrow & & \downarrow{\scriptstyle m'} & & \downarrow{\scriptstyle m''} \\ C & \xleftarrow{\quad f' \quad} & D & \xrightarrow{\quad g' \quad} & E \end{array}$$

in **PChem** such that the top span $A \leftarrow K \rightarrow B$ is a reaction scheme, $m, m', m'' \in \mathcal{M}$ are matchings, and C and E are chemical graphs.

Example 5.4. The following reaction (glucose phosphorylation) is an instance of the reaction scheme in Example 5.2; we have labelled the vertices in the images of matchings on both sides (the sets U_A and U_B), and we use the convention from chemistry where an unlabelled vertex is a carbon atom with an appropriate number of hydrogen atoms attached:



Note that the graphs appearing as the boundaries of the span of the reaction scheme are the disjoint unions of the valence completion and charge decomposition of the graphs in the above reaction (cf. Examples 4.11 and 4.13). We make this observation precise in Proposition 5.6.

Theorem 5.5. *Let C be a chemical graph. Given a matching and a reaction scheme as below left, the diagram can be uniquely completed to the reaction instance on the right.*

$$\begin{array}{ccccc} A & \xleftarrow{f} & K & \xrightarrow{g} & B \\ {\scriptstyle m}\downarrow & & & & \\ C & & & & \end{array} \qquad \begin{array}{ccccc} A & \xleftarrow{f} & K & \xrightarrow{g} & B \\ {\scriptstyle m}\downarrow & & \downarrow{\scriptstyle m'} & & \downarrow{\scriptstyle m''} \\ C & \xleftarrow{f'} & D & \xrightarrow{g'} & E \end{array}$$

Proof. The pre-chemical graph D can be constructed as the pushout complement (Corollary 4.25). The pre-chemical graph E is then obtained as the pushout, which exists by Proposition 4.22. □

A reaction instance can be presented in a more concrete (yet equivalent) way, which involves mappings that are not morphisms in **PChem**.

Proposition 5.6. *Let C and E be chemical graphs. The data of a reaction instance $C \to E$ can be equivalently (up to an isomorphism) presented as a tuple (U_C, U_E, b, i) where $U_C \subseteq V_C$ and $U_E \subseteq V_E$ are matchable subsets with equal net charge, $b : \mathtt{Chem}(U_C) \to \mathtt{Chem}(U_E)$ is a bijection preserving the atom labels, and $i : V_C \setminus U_C \to V_E \setminus U_E$ is an isomorphism of pre-chemical graphs such that for all $u \in \mathtt{Chem}(U_A)$ and $a \in V_A \setminus U_A$ we have $m_A(u, a) = m_B(bu, ia)$.*

Proof. Given a reaction instance

$$\begin{array}{ccccc} A & \xleftarrow{f} & K & \xrightarrow{g} & B \\ {\scriptstyle m}\downarrow & & \downarrow{\scriptstyle m'} & & \downarrow{\scriptstyle m''} \\ C & \xleftarrow{f'} & D & \xrightarrow{g'} & E \end{array},$$

we obtain the desired tuple as $(m(A), m''(B), b, i)$, where b and i are both appropriate restrictions of $g'(f')^{-1}$.

Conversely, given a tuple as in the statement of the proposition, we obtain the following reaction instance

$$\begin{array}{ccccc} U_C^* & \hookleftarrow & K & \xrightarrow{\hat{b}} & U_E^* \\ \downarrow{\scriptstyle m} & & \downarrow & & \downarrow{\scriptstyle m'} \\ C & \hookleftarrow & D & \hookrightarrow & E \end{array},$$

where m and m' are the matchings defined from matchable subsets in the proof of Proposition 4.16. In order to define the graph K, we first note that the bijection $b : \texttt{Chem}(U_C) \to \texttt{Chem}(U_E)$ induces an atom label preserving bijection

$$\bar{b} : m^{-1}(\texttt{Chem}(C)) \cap \alpha(U_C^*) \to (m')^{-1}(\texttt{Chem}(E)) \cap \alpha(U_E^*)$$

as follows. For every $c \in \texttt{Chem}(U_C) \cap m(\alpha(U_C^*))$, we define $m^{-1}(c) \to (m')^{-1}(bc)$ by the following procedure:

1. Let $N_c \coloneqq a_1, \ldots, a_n$ and $C_c \coloneqq b_1, \ldots, b_m$ be lists all the neutral and charged vertices of $m^{-1}(c)$, respectively. Similarly, let $N_{bc} \coloneqq c_1, \ldots, c_q$ and $C_{bc} \coloneqq d_1, \ldots, d_p$ be lists all the neutral and charged vertices of $(m')^{-1}(bc)$. Note that we have $n + m = p + q$.
2. For each $i = 1, \ldots, n$, let n_i be the unique covalent neighbour of a_i. If there is a $c_j \in N_{bc}$ such that $c_j \in \texttt{CN}_{U_E^*}(bn_i)$, define $a_i \mapsto c_j$, and remove a_i from N_c and c_j from N_{bc}.
3. For each $i = 1, \ldots, m$, if there is a $d_j \in C_{bc}$ with the same charge as b_i such that $b\left(\texttt{IN}_{U_C^*}(b_i)\right) = \texttt{IN}_{U_E^*}(d_j)$, define $b_i \mapsto d_j$, and remove b_i from C_c and d_j from C_{bc}.
4. The remaining vertices in N_c and C_c may be mapped to any remaining vertices in N_{bc} and C_{bc} (as long as we have a bijection).

Next, define the following subset of α-vertices $A \subseteq V_{U_C}$. For every $u \in \texttt{Chem}(U_C)$, let $A_u^{\texttt{c}}$ be any subset of $\texttt{CN}_C(u) \cap \alpha(U_C)$ of size

$$\min\left(\left|\texttt{CN}_C(u) \cap \alpha(U_C)\right|, \left|\texttt{CN}_E(bu) \cap \alpha(U_E)\right|\right).$$

Let $A_u^{\texttt{i}-}$ be any subset of $\texttt{IN}_C(u) \cap \alpha(U_C) \cap \texttt{Crg}^-\left(U_C\right)$ of size

$$\min\left(\left|\texttt{IN}_C(u) \cap \alpha(U_C) \cap \texttt{Crg}^-\left(U_C\right)\right|, \left|\texttt{IN}_E(bu) \cap \alpha(U_E) \cap \texttt{Crg}^-\left(U_E\right)\right|\right).$$

Similarly, let $A_u^{\texttt{i}+}$ be any subset of positive ionic α-neighbours of u, whose size is that of positive ionic α-neighbours of u or positive ionic α-neighbours of $b(u)$, whichever is smaller. Let A^- be any subset of isolated negative α-vertices in U_C, whose size is the smaller of isolated negative α-vertices in U_C and isolated negative α-vertices in U_E. The set A^+ is defined similarly, but for isolated positive α-vertices. We then define

$$A \coloneqq A^- \cup A^+ \cup \bigcup_{u \in \texttt{Chem}(U_C)} A_u^{\texttt{c}} \cup A_u^{\texttt{i}-} \cup A_u^{\texttt{i}+}.$$

Note that there is an injection $\iota : A \to U_E^*$ that takes each vertex $a \in A$ to an α-vertex $\iota(a)$ with the same charge, such that the neighbour of a is mapped by b to the neighbour of $\iota(a)$. We denote by

$$\hat{b} : \texttt{Chem}(U_C^*) \cup \left(m^{-1}(\texttt{Chem}(C)) \cap \alpha(U_C^*)\right) \cup A \to U_E^*$$

the map that acts as $b + \bar{b} + \iota$ on the disjoint summands. We now define the graph K as follows:

- the vertex set is $V_K \coloneqq \texttt{Chem}(U_C^*) \cup \big(m^{-1}(\texttt{Chem}(C)) \cap \alpha(U_C^*)\big) \cup A$,
- the atom labels are the same as in U_C^*,
- for every $k \in V_K$, if $\tau_{U_C^*}^{\texttt{Crg}}(k) = \tau_{U_E^*}^{\texttt{Crg}}(\hat{b}k) =: n$, then define $\tau_K^{\texttt{Crg}}(k) \coloneqq n$, otherwise define $\tau_K^{\texttt{Crg}}(k) \coloneqq 0$,
- for all $k, t \in V_K$, if $m_{U_C^*}(k,t) = m_{U_E^*}(\hat{b}k, \hat{b}t) =: n$, then define $m_K(k,t) \coloneqq n$, otherwise define $m_K(k,t) \coloneqq 0$.

The graph D is then constructed as the pushout complement for both m and m'. □

Proposition 5.6 motivates the following definition:

Definition 5.7 (Category of reactions)**.** We denote by **React** the *category of reactions*, whose

- objects are chemical graphs,
- morphisms $A \to B$ are tuples (U_A, U_B, b, i), where
  - $U_A \subseteq V_A$ and $U_B \subseteq V_B$ are subsets with $\texttt{Net}\,(U_A) = \texttt{Net}\,(U_B)$,
  - $b : \texttt{Chem}(U_A) \to \texttt{Chem}(U_B)$ is a bijection preserving the atom labels,
  - $i : V_A \setminus U_A \to V_B \setminus U_B$ is an isomorphism of pre-chemical graphs,

  such that for all $u \in \texttt{Chem}(U_A)$ and $a \in V_A \setminus U_A$ we have
  $$m_A(u, a) = m_B(bu, ia),$$
- the composition of $(U_A, U_B, b, i) : A \to B$ and $(W_B, W_C, c, j) : B \to C$ is given by
  $$(Z_A, Z_C, (c + j)(b + i), ji) : A \to C,$$
  where $Z_A \coloneqq U_A \cup i^{-1}(W_B \setminus U_B))$ and $Z_C \coloneqq W_C \cup j(U_B \setminus W_B))$,
- for a molecular graph A, the identity is given by $(\varnothing, \varnothing, !, \mathrm{id}_A)$, where ! is the unique endomorphism on the empty set.

Note that the composition in **React** is *not* the composition in the usual category of partial bijections: instead, it crucially relies on the fact that there is an isomorphism between the unchanged parts of the graph. The category **React** has a dagger structure: the dagger of $(U_A, U_B, b, i) : A \to B$ is given by $(U_B, U_A, b^{-1}, i^{-1}) : B \to A$. Given a morphism $r \in \textbf{React}$, we will denote its dagger by $\overline{r}$.

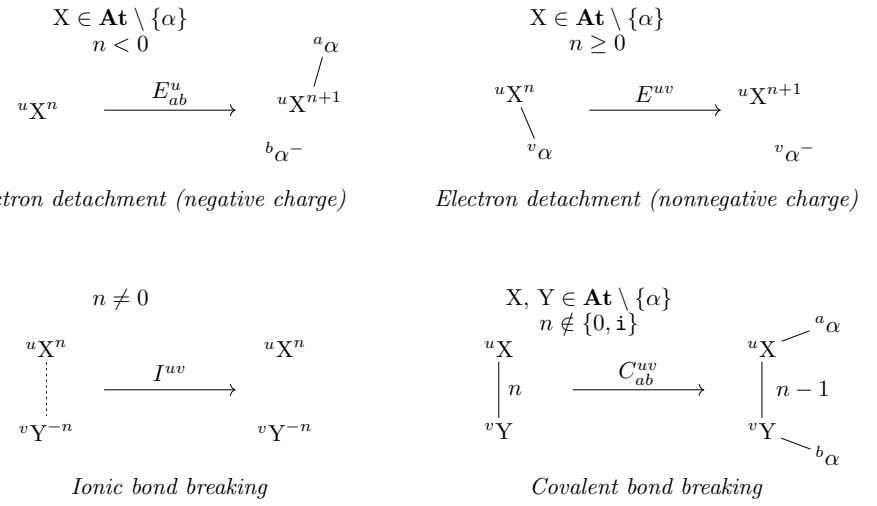
Remark 5.8. Note that the morphisms in **React** are slightly more general than tuples arising from reaction schemes in Proposition 5.6. Namely, we do not require the subsets in a morphism in **React** to be matchable. This generalisation is, however, merely technical, as we can always extend any subset to the smallest matchable one while keeping the same information about which bonds and charges are reconnected and exchanged. The reason for allowing more morphisms in **React** is to obtain an exact correspondence with the disconnection rules (Section 6), which operate only on single edge or vertex at a time, and hence are unable to capture global conditions, such as being matchable.

6. The Category of Disconnection Rules

A *disconnection rule* is a partial endofunction on the set of chemical graphs. We define four classes of disconnection rules, all of which have a clear chemical significance: two versions of *electron detachment*, *ionic bond breaking* and *covalent bond breaking*. These local chemical transformations are our third perspective on chemical processes. The reader may want to check Figure 5 before Definition 6.1 below, as it gives an intuitive explanation of our approach.

For the purposes of mathematical precision, our set of four disconnection rules is more fine-grained than what one would see in a typical textbook on retrosynthesis, where movement of electrons is usually implicitly modelled in the same step as disconnecting a bond, rather than including electron detachment as a separate step (see, for instance, the discussion on the choice of polarity in [1, p. 9]).

We treat the disconnection rules as syntax, which generate the *terms* (Definition 6.2), whose equivalence classes under the equations of Figure 7 form the morphisms in the *disconnection category* (Definition 6.5). The payoff such a syntactic presentation is an axiomatic view of chemical reactions: in Section 7, we construct a functor from the disconnection category to the category of reactions, and show that every reaction can be represented as a sequence of disconnection rules in an essentially unique way.



Electron detachment (negative charge) *Electron detachment (nonnegative charge)*

Ionic bond breaking *Covalent bond breaking*

Figure 5: The four disconnection rules.

Definition 6.1 (Disconnection rules)**.** Let $u, v, a, b \in \mathbf{VN}$ be pairwise distinct vertex names. Let $U \in \{u, uv\}$ and $D \in \{\varnothing, ab\}$ range over the specified lists of vertex names. The four *disconnection rules* are defined by the tables in Figure 6 as follows: a chemical graph A is in the domain of d_D^U if $U \subseteq V_A$ but $D \cap V_A = \varnothing$, and the additional conditions of the first column (top table) hold; the output chemical graph $d(A)$ has the vertex set $V_A \cup D$, and the labelling functions on $U \cup D$ are defined by the remaining columns (vertex labelling in the top table, edge labelling in the bottom table), while the labelling functions agree with those of A on $V_A \setminus U$.

d_D^U	$A \in dom(d)$	$\tau^{\mathtt{Crg}}_{d(A)}(u)$	$\tau^{\mathtt{Crg}}_{d(A)}(v)$	$\tau_{d(A)}(a)$	$\tau_{d(A)}(b)$
E^u_{ab}	$u \in \mathtt{Chem}(A)$ $u \in \mathtt{Crg}^-(A)$	$\tau^{\mathtt{Crg}}_A(u) + 1$	N/A	$(\alpha, 0)$	$(\alpha, -1)$
E^{uv}	$u \in \mathtt{Chem}(A)$ $u \notin \mathtt{Crg}^-(A)$ $v \in \alpha(A)$ $m_A(u,v) = 1$	$\tau^{\mathtt{Crg}}_A(u) + 1$	-1	N/A	N/A
I^{uv}	$m_A(u,v) = \mathtt{i}$ $u \in \mathtt{Crg}^+(A)$ $v \in \mathtt{Crg}^-(A)$	$\tau^{\mathtt{Crg}}_A(u)$	$\tau^{\mathtt{Crg}}_A(v)$	N/A	N/A
C^{uv}_{ab}	$u, v \in \mathtt{Chem}(A)$ $m_A(u,v) \notin \{0, \mathtt{i}\}$	$\tau^{\mathtt{Crg}}_A(u)$	$\tau^{\mathtt{Crg}}_A(v)$	$(\alpha, 0)$	$(\alpha, 0)$

d_D^U	$m_{d(A)}(u,v)$	$m_{d(A)}(u,a)$	$m_{d(A)}(v,b)$
E^u_{ab}	N/A	1	N/A
E^{uv}	0	N/A	N/A
I^{uv}	0	N/A	N/A
C^{uv}_{ab}	$m_A(u,v) - 1$	1	1

Figure 6: The disconnection rules defined as partial functions.

Note that the disconnection rules look a lot like (a subset of) reaction schemes (Definition 5.1): indeed, each disconnection rule can be realised as a collection of reaction schemes. In Section 5, we already saw that reactions arise from reaction schemes (Proposition 5.6). In Section 7, we shall strengthen this result by showing that the disconnection rules generate and axiomatise all the reactions as morphisms in **React**.

We observe that each disconnection rule is injective (as a partial function), and hence has an inverse partial function. We use the disconnection rules to define the *terms*, which will be used to define the morphisms in the disconnection category.

Definition 6.2 (Terms)**.** The set of *terms* with types is generated by the following recursive procedure:

- for every chemical graph A, let $\mathrm{id} : A \to A$ be a term,
- for every chemical graph A and every $u \in V_A$, let $S^u : A \to A$ be a term,
- for every chemical graph A, every $u \in \alpha(A)$ and every $v \in \mathbf{VN}$ such that $v \notin V_A \setminus \{u\}$, let $R^{u \mapsto v} : A \to A(u \mapsto v)$ be a term,
- for every disconnection rule d and every chemical graph A in the domain of d, both $d : A \to d(A)$ and $\bar{d} : d(A) \to A$ are terms,
- if $\mathtt{t} : A \to B$ and $\mathtt{s} : B \to C$ are terms, then $\mathtt{t}; \mathtt{s} : A \to C$ is a term.

The first and the fifth items take care of the usual categorical structure, while the terms S^u generated by the second item correspond to "touching" the vertex u without changing the structure of the graph, and the terms $R^{u \mapsto v}$ rename an existing α-vertex u into a fresh vertex v.

We refer to the terms of the form $d : A \to B$ and $\bar{d} : B \to A$ generated by the fourth item as *disconnections* and *connections*, respectively. More specifically, we use the symbols $E^{<0}$, $E^{\geq 0}$, I and C to denote the disconnections corresponding to the specific disconnection rules, and similarly the symbols $\bar{E}^{<0}$, $\bar{E}^{\geq 0}$, $\bar{I}$ and $\bar{C}$ refer to the corresponding connections. Similarly, S and R refer to the terms generated by the second and third items. The same letters in the typewriter type font ($\mathtt{E}^{<0}$, $\mathtt{E}^{\geq 0}$, $\mathtt{I}$, $\mathtt{C}$, $\bar{\mathtt{E}}^{<0}$, $\bar{\mathtt{E}}^{\geq 0}$, $\bar{\mathtt{I}}$, $\mathtt{S}$ and $\mathtt{R}$) are used to denote a sequence of terms of the corresponding kind.

Let us define the endofunction $\overline{()}$ on terms by the following recursion:

- $(\mathrm{id} : A \to A) \mapsto (\mathrm{id} : A \to A)$,
- $(S^u : A \to A) \mapsto (S^u : A \to A)$,
- $(R^{u \mapsto v} : A \to A(u \mapsto v)) \mapsto (R^{v \mapsto u} : A(u \mapsto v) \to A)$,
- $(d : A \to B) \mapsto \big(\bar{d} : B \to A\big)$,
- $\big(\bar{d} : A \to B\big) \mapsto (d : B \to A)$,
- $\overline{\mathtt{t}; \mathtt{s}} \coloneqq \overline{\mathtt{s}}; \overline{\mathtt{t}}$.

For defining equations, it will be useful to allow untyped terms: the equations (Figure 7) capture interactions between local graph transformations (i.e. the disconnection rules), so that the same equation should hold for a whole class of chemical graphs.

Definition 6.3 (Untyped terms, well-typedness)**.** An *untyped term* is an element of the free monoid on the set

$$\{\mathrm{id}, S^u, R^{a \mapsto b}, E^{ua}, E^u_{ab}, C^{uv}_{ab}, I^{uv}, \bar{E}^{ua}, \bar{E}^u_{ab}, \bar{C}^{uv}_{ab}, \bar{I}^{uv} : u, v, a, b \in \mathbf{VN}\},$$

where we use the symbol ; to indicate the multiplication of the monoid.

Given an untyped term $\mathtt{t}$ and chemical graphs A and B, we say that the expression $\mathtt{t} : A \to B$ is *well-typed* if it is in fact a term, that is, if it can be constructed using the recursive procedure of Definition 6.2.

We define the binary relation $\leq$ on the set of untyped terms by letting $\mathtt{t} \leq \mathtt{s}$ if whenever $\mathtt{t} : A \to B$ is well-typed, then so is $\mathtt{s} : A \to B$.

The endofunction $\overline{()}$ on the untyped terms is defined in exactly the same way as for the terms with types, simply ignoring the types. Note that $\mathtt{t} : A \to B$ is well-typed if and only if $\overline{\mathtt{t}} : B \to A$ is. Moreover, observe that $\leq$ defines a preorder on the untyped terms. Consequently, we have $\mathtt{t} \leq \mathtt{s}$ if and only if $\overline{\mathtt{t}} \leq \overline{\mathtt{s}}$.

Given an untyped term $\mathtt{t}$, there are either no chemical graphs such that $\mathtt{t} : A \to B$ is well-typed, or there are infinitely many such graphs. The latter case is the reason for introducing the untyped terms: we want certain equalities to hold *whenever* both sides are well-typed.

Definition 6.4 (Term equality)**.** Let $\approx$ be an equivalence relation on the set of untyped terms. This induces the equivalence relation $\equiv$ on the set of terms as follows: for two terms $\mathtt{t}, \mathtt{s} : A \to B$ with the same type, we let $\mathtt{t} \equiv \mathtt{s}$ if either $\mathtt{t} \approx \mathtt{s}$ or $\overline{\mathtt{t}} \approx \overline{\mathtt{s}}$ as untyped terms.

Given an equivalence relation $\approx$ on the untyped terms, let us introduce the following shorthand binary relations on the untyped terms:

- $\mathtt{t} \lesssim \mathtt{s}$ if $\mathtt{t} \approx \mathtt{s}$ and $\mathtt{t} \leq \mathtt{s}$,
- $\mathtt{t} \simeq \mathtt{s}$ if $\mathtt{t} \lesssim \mathtt{s}$ and $\mathtt{s} \lesssim \mathtt{t}$.

Definition 6.5 (Disconnection category)**.** The *disconnection category* **Disc** has as objects the chemical graphs. The set of morphisms $\mathbf{Disc}(A, B)$ is given by the terms of type $A \to B$, subject to the usual associativity and unitality equations of a category, together with the identities $\equiv$ induced (in the sense of Definition 6.4) by the equivalence relation defined in Figure 7.

Note that the assignment $\overline{()} : \mathbf{Disc} \to \mathbf{Disc}$ is functorial, thus making **Disc** a dagger category [39, 40].

Proposition 6.6. *The following identities are derivable in* **Disc***:*

$$d^U_{D[a]}; S^a \simeq d^U_{D[a]}, \tag{35}$$

$$\bar{d}^U_{ab}; d^U_{cd} \approx S^U; R^{a\mapsto j}; R^{b\mapsto d}; R^{j\mapsto c}, \tag{36}$$

$$R^{z\mapsto c}; R^{w\mapsto d}; d^U_{ab} \approx R^{z\mapsto a}; R^{w\mapsto b}; d^U_{cd}, \tag{37}$$

$$R^{z\mapsto c}; d^U_{ab} \approx R^{z\mapsto a}; d^U_{cb}, \tag{38}$$

$$R^{w\mapsto d}; d^U_{ab} \approx R^{w\mapsto b}; d^U_{ad}, \tag{39}$$

$$d^U_{ab}; d^U_{cd} \simeq d^U_{ad}; d^U_{cb}. \tag{40}$$

Proof. We compute (35) by applying equations (3) and (9):

$$d^U_{D[a]}; S^a \simeq d^U_{D[a]}; R^{a\mapsto a} \simeq d^U_{D[a]}.$$

Equality (36) is derived as follows:

$$\begin{aligned}
\bar{d}^{uv}_{ab}; d^{uv}_{cd} &\approx d^{uv}_{ij}; \bar{d}^{uv}_{ab}; R^{i\mapsto c}; R^{j\mapsto d} && \text{(by (10))}\\
&\approx S^u; S^v; R^{a\mapsto i}; R^{b\mapsto j}; R^{i\mapsto c}; R^{j\mapsto d} && \text{(by (11))}\\
&\approx S^u; S^v; R^{a\mapsto j}; R^{b\mapsto d}; R^{j\mapsto c}. && \text{(by (2) and (1))}
\end{aligned}$$

For (37), we first use (36) to get

$$d^{uv}_{cd}; \bar{d}^{uv}_{zw}; d^{uv}_{ab} \simeq d^{uv}_{cd}; S^u; S^v; R^{z\mapsto a}; R^{w\mapsto b},$$

where on the right-hand side we used (2) and (1). Observing that (11) applies on the left-hand side, and simplifying using (5), (21) and (7), we obtain precisely (37). Identities (38) and (39) are derived similarly, by letting $w = d$ and $z = c$, respectively.

Identity (40) is derived as follows:

$$\begin{aligned}
d^U_{ab}; d^U_{cd} &\simeq d^U_{ad}; R^{d\mapsto b}; d^U_{cd} && \text{(by (9))}\\
&\simeq d^U_{ad}; R^{d\mapsto d}; d^U_{cb} && \text{(by (39))}\\
&\simeq d^U_{ad}; d^U_{cb}. && \text{(by (9))}
\end{aligned}$$

□

$$R^{u\mapsto z};R^{z\mapsto w}\lesssim R^{u\mapsto w} \quad (1)$$
$$R^{u\mapsto z};R^{v\mapsto w}\approx R^{v\mapsto w};R^{u\mapsto z} \quad (2)$$
$$R^{u\mapsto u}\lesssim S^u \quad (3)$$
$$R^{b\mapsto z};R^{a\mapsto b}\approx S^b;R^{a\mapsto z} \quad (4)$$
$$R^{u\mapsto v};S^w\approx S^w;R^{u\mapsto v} \quad (5)$$
$$R^{u\mapsto v};S^v\simeq S^u;R^{u\mapsto v}\simeq R^{u\mapsto v} \quad (6)$$

$$R^{u\mapsto v};d^U_D\approx d^U_D;R^{u\mapsto v} \quad (7)$$
$$R^{u\mapsto v};E^{wv}\simeq E^{wu};R^{u\mapsto v} \quad (8)$$
$$d^U_{D[u]};R^{u\mapsto v}\simeq d^U_{D[v/u]} \quad (9)$$
$$d^{U'}_{ij};\bar{h}^U_{ab};R^{i\mapsto c};R^{j\mapsto d}\lesssim \bar{h}^U_{ab};d^{U'}_{cd} \quad (10)$$

$$d^U_{ab};\bar{d}^U_{cd}\approx S^U;R^{c\mapsto a};R^{d\mapsto b} \quad (11)$$
$$d^U_{ab};\bar{d}^U_{cb}\approx S^U;R^{c\mapsto a} \quad (12)$$
$$d^U_{ab};\bar{d}^U_{ad}\approx S^U;R^{d\mapsto b} \quad (13)$$
$$d^U_D;\bar{d}^U_D\lesssim S^U \quad (14)$$
$$\bar{d}^U_D;d^U_D\lesssim S^U;S^D \quad (15)$$
$$E^{ua};\bar{E}^{ub}\approx S^u;R^{a\mapsto z};R^{b\mapsto a};R^{z\mapsto b} \quad (16)$$
$$\bar{d}^{uv};d^{wz}\approx d^{wz};\bar{d}^{uv} \quad (17)$$

$$S^u;S^v\simeq S^v;S^u \quad (18)$$
$$S^u;S^u\simeq S^u \quad (19)$$
$$S^u;d^U_D\lesssim d^U_D;S^u \quad (20)$$
$$d^{U[v]}_D;S^v\simeq d^{U[v]}_D \quad (21)$$
$$C^{uv}_{ab}\simeq C^{vu}_{ba} \quad (22)$$

$$d^U_D;d^{U'}_{D'}\simeq d^{U'}_{D'};d^U_D \quad (23)$$
$$C^{uv}_{ab};I^{wz}\simeq I^{wz};C^{uv}_{ab} \quad (24)$$
$$E^u_{ab};I^{wz}\lesssim I^{wz};E^u_{ab} \quad (25)$$
$$E^{uv};I^{wz}\lesssim I^{wz};E^{uv} \quad (26)$$
$$\bar{E}^{uv};I^{wz}\lesssim I^{wz};\bar{E}^{uv} \quad (27)$$
$$\bar{E}^u_{ab};I^{wz}\lesssim I^{wz};\bar{E}^u_{ab} \quad (28)$$
$$\bar{C}^{uv}_{ab};I^{wz}\lesssim I^{wz};\bar{C}^{uv}_{ab} \quad (29)$$
$$E^u_{ab};C^{wz}_{cd}\simeq C^{wz}_{cd};E^u_{ab} \quad (30)$$
$$E^{uv};C^{wz}_{cd}\lesssim C^{wz}_{cd};E^{uv} \quad (31)$$
$$\bar{E}^{uv};C^{wz}_{cd}\simeq C^{wz}_{cd};\bar{E}^{uv} \quad (32)$$
$$E^{uv};E^w_{cd}\lesssim E^w_{cd};E^{uv} \quad (33)$$
$$\bar{E}^{uv};E^w_{cd}\simeq E^w_{cd};\bar{E}^{uv} \quad (34)$$

Figure 7: The equivalence relation $\approx$ inducing the identities in the disconnection category. Here d and h range over $\{E, C, I\}$, while S^U stands for the sequence $S^u;S^w$ if $U = uw$. Given vertex names $a, b \in \mathbf{VN}$, the notation $D[a]$ means a occurs in D, and $D[b/a]$ means the occurrence of a in D is replaced with b. Note that we use the shorthand relations $\lesssim$ and $\simeq$: these are strictly speaking not part of the definition, but are used to provide the extra information of when well-typedness of one side of an identity implies well-typedness of the other.

6.1. Normal Form

In this subsection, we define a normal form (Definition 6.11), and show that every term is equal to a term in a normal form under the equalities of **Disc** (Proposition 6.12). We also identify a class of syntactic manipulations of terms in a normal form (Definition 6.13) that both keep the normal form and preserve equality (Lemma 6.14). These results are used in the next section to prove completeness.

Definition 6.7 (*ICE*-form)**.** We say that a term is in an *ICE-form* if it is either an identity term, or if it has the following structure:

$$\mathtt{I};\mathtt{C};\mathtt{E}^{<0};\mathtt{E}^{\geq 0};\bar{\mathtt{E}}^{\geq 0};\bar{\mathtt{E}}^{<0};\bar{\mathtt{C}};\bar{\mathtt{I}};\mathtt{R};\mathtt{S},$$

where every letter is a sequence of generating terms of the corresponding kind.

Proposition 6.8. *Any term is equal to a term in an ICE-form.*

Proof sketch. The proof proceeds by repeated inductions: one first shows that all I-terms can always be commuted to the left, then that all C-terms can be commuted to the left of anything that is not an I-term, and so on. We give the full details of the induction in the Appendix (Appendix A.1-Appendix A.18). □

Definition 6.9 (Renaming form)**.** A well-typed sequence of renaming terms $\mathtt{R} : H \to G$ is in a *renaming form* if there are sets $A = \{a_1, \ldots, a_n\}$, $B = \{b_1, \ldots, b_n\}$, $C = \{c_1, \ldots, c_m\}$ and $D = \{d_1, \ldots, d_m\}$ of vertex names such that

(1) $\mathtt{R}$ can be split into two sequences $\mathtt{R} = \mathtt{A};\mathtt{B}$ with

$$\mathtt{A} = R^{a_1 \mapsto b_1};\ldots;R^{a_n \mapsto b_n} \quad \text{and} \quad \mathtt{B} = R^{c_1 \mapsto d_1};\ldots;R^{c_m \mapsto d_m},$$

where $\mathtt{B}$ can be possibly empty,

(2) $A \cap B = \varnothing$,

(3) $C \subseteq B$,

(4) $D \subseteq A$,

(5) if $c_i \in C$ and $b_j \in B$ is the unique element such that $b_j = c_i$, then $\mathtt{N}_H(a_j) \neq \mathtt{N}_H(d_i)$.

Lemma 6.10. *Any well-typed sequence of renaming terms is equal to a term* $\mathtt{R};\mathtt{S}$*, where* $\mathtt{R} = \mathtt{A};\mathtt{B}$ *is in a renaming form and* $\mathtt{S}$ *is a sequence of S-terms.*

Proof. The idea is that if a vertex a is to be renamed to b, then $a \in A$, and we have two cases: (1) b does not already occur in the original chemical graph, and (2) b does occur in the original graph. If (1), then $R^{a \mapsto b} \in \mathtt{A}$ and $b \in B \setminus C$. If (2), then we first rename a using some "dummy" name c, so that $R^{a \mapsto c} \in \mathtt{A}$, $R^{c \mapsto b} \in \mathtt{B}$, $c \in C$ and $b \in D$. Note that condition (4) of the renaming form is satisfied, as b must itself be renamed in order for the vertex name become free. Any term of the form $R^{a \mapsto a}$ is replaced by S^a. The formal proof proceeds by induction on the length of the original sequence.

The term $R^{a \mapsto b}$ is equal to S^a if $a = b$, or is already in a renaming form by taking $A = \{a\}$, $B = \{b\}$ and $C = D = \varnothing$ if $a \neq b$.

Suppose that the statement of the lemma holds for all sequences of renaming terms of length at most n. Let $\mathtt{R}$ be such a sequence of length n such that $\mathtt{R}; R^{a\mapsto b}$ is well-typed. By the inductive hypothesis, we may assume that $\mathtt{R} = \mathtt{A};\mathtt{B};\mathtt{S}$ where $\mathtt{A};\mathtt{B}$ is a renaming form with vertex name sets A, B, C and D as in Definition 6.9. Using the equations for S- and R-terms, we may commute $R^{a\mapsto b}$ past $\mathtt{S}$, possibly changing the vertex name a, so that it suffices to show that the lemma holds for $\mathtt{A};\mathtt{B}; R^{a\mapsto b}$. If $a = b$, the sequence is equal to $\mathtt{A};\mathtt{B}; S^a$ and we are done; hence assume that $a \neq b$. Note that it follows that $a \notin A \setminus D$ and $a \notin C$, as every vertex name in $A \setminus D$ or C is removed, without being reintroduced. Similarly, we have that $b \notin D$ and $b \notin B \setminus C$. Moreover, if $b \in C$, rename the occurrence of b in both $\mathtt{A}$ and $\mathtt{B}$ with a fresh vertex name, updating the sets C and B accordingly. Thus we may assume that $b \notin B$. The remaining cases are as follows.

Case 1: $a \notin A \cup B$.

Subcase 1.1: $b \notin A$. We rewrite the term to $\mathtt{A}; R^{a\mapsto b}; \mathtt{B}$ and update the sets $A \mapsto A \cup \{a\}$ and $B \mapsto B \cup \{b\}$.

Subcase 1.2: $b \in A$. It follows that $b \in A \setminus D$, so that $R^{b\mapsto z} \in \mathtt{A}$ and a, b do not appear in $\mathtt{B}$. If $\mathtt{N}(a) \neq \mathtt{N}(b)$, let c be a fresh vertex name. We rewrite the term to $\mathtt{A}; R^{a\mapsto c}; R^{c\mapsto b}; \mathtt{B}$ and update the sets $A \mapsto A \cup \{a\}$, $B \mapsto B \cup \{c\}$, $C \mapsto C \cup \{c\}$ and $D \mapsto D \cup \{b\}$. If $\mathtt{N}(a) = \mathtt{N}(b)$, we use equation (4) to rewrite $R^{b\mapsto z}; R^{a\mapsto b}$ to $S^b; R^{a\mapsto z}$, which reduces the number of R-terms to n, so that the inductive hypothesis applies.

Case 2: $a \in A$. It follows that $a \in D$. Now $R^{a\mapsto b}$ commutes with all other terms in $\mathtt{B}$ except for the unique term $R^{c_i\mapsto d_i}$ where $d_i = a$. But $R^{c_i\mapsto a}; R^{a\mapsto b} \equiv R^{c_i\mapsto b}$, which reduces the length of the sequence to n, so it is has a renaming form by the inductive hypothesis.

Case 3: $a \in B$. It follows that $a \in B \setminus C$.

Subcase 3.1: $b \notin A$. Now $R^{a\mapsto b}$ commutes with all the terms in $\mathtt{B}$, and with all other terms in $\mathtt{A}$ except for the unique term $R^{a_i\mapsto b_i}$ where $b_i = a$. But $R^{a_i\mapsto a}; R^{a\mapsto b} \equiv R^{a_i\mapsto b}$, which reduces the length of the sequence to n, so it is has a renaming form by the inductive hypothesis.

Subcase 3.2: $b \in A$. It follows that $b \in A \setminus D$. Now $R^{a\mapsto b}$ commutes with all the terms in $\mathtt{B}$, and with all other terms in $\mathtt{A}$ except for the terms $R^{a_i\mapsto a}$ and $R^{b\mapsto b_j}$. There are two options: (1) $a_i = b$ and $b_j = a$, so that these are the same term, (2) the terms are distinct, in which case they commute. In both cases, we use the substitution $R^{a_i\mapsto a}; R^{a\mapsto b} \equiv R^{a_i\mapsto b}$ to reduce the length of the sequence, so that the inductive hypothesis applies.

This completes the induction. □

A term is said to be in an *ICER-form* if it is in an *ICE*-form whose sequence of renaming terms is in a renaming form (or is empty).

Definition 6.11 (Normal form)**.** Let

$$\mathtt{t} = \mathtt{I};\mathtt{C};\mathtt{E}^{<0};\mathtt{E}^{\geq 0};\bar{\mathtt{E}}^{\geq 0};\bar{\mathtt{E}}^{<0};\bar{\mathtt{C}};\bar{\mathtt{I}};\mathtt{A};\mathtt{B};\mathtt{S}$$

be a term in an *ICER*-form. Let us denote the sets of vertex names in the renaming form by $A_\mathtt{t}$, $B_\mathtt{t}$, $C_\mathtt{t}$ and $D_\mathtt{t}$. Let us additionally define the following sets of vertex names occurring in $\mathtt{t}$:

- $D^{add}_\mathtt{t} \coloneqq \left\{a \in \mathbf{VN} : d^U_{D[a]} \in \mathtt{t}\right\}$ – the vertex names appearing as subscripts in the disconnections,

- $D^{remove}_{\mathtt{t}} \coloneqq \left\{a \in \mathbf{VN} : \bar{d}^{U}_{D[a]} \in \mathtt{t}\right\}$ – the vertex names appearing as subscripts in the connections,
- $U_{\mathtt{t}} \coloneqq \left\{v \in \mathbf{VN} : d^{U[v]}_{D} \in \mathtt{t} \text{ or } \bar{d}^{U[v]}_{D} \in \mathtt{t}\right\}$ – the vertex names appearing as superscripts of the (dis)connections,
- $S_{\mathtt{t}} \coloneqq \{u \in \mathbf{VN} : S^u \in \mathtt{t}\}$ – the vertex names appearing in the S-terms.

We say that a term $\mathtt{t}$ is in a *normal form* if it is in an $ICER$-form as above, and additionally the following conditions hold:

(1) for every $u \in S_{\mathtt{t}}$, the term S^u occurs in $\mathtt{t}$ exactly once,

(2) $(U_{\mathtt{t}} \cup A_{\mathtt{t}} \cup B_{\mathtt{t}}) \cap S_{\mathtt{t}} = \varnothing$,

(3) $D^{add}_{\mathtt{t}} \setminus D^{remove}_{\mathtt{t}} \subseteq A_{\mathtt{t}} \setminus D_{\mathtt{t}}$,

(4) $D^{add}_{\mathtt{t}} \cap B_{\mathtt{t}} = \varnothing$,

(5) if a connection $\bar{d}^{U}_{D[a]} : A \to B$ and a renaming term $R^{z \mapsto a}$ both occur, then A is not in the domain of $\bar{d}^{U}_{D[z/a]}$,

(6) if $d \neq I$ and a disconnection d^{U}_{D} occurs in $\mathtt{t}$, then the connections $\bar{d}^{U}_{F}$ and $\bar{d}^{U^r}_{F}$ do not occur in $\mathtt{t}$ for any F (here U^r denotes the reverse string),

(7) if the disconnection E^{uv} occurs in $\mathtt{t}$, then for any vertex name $w \in \mathbf{VN}$, the connection $\bar{E}^{uw}$ does not occur in $\mathtt{t}$,

(8) if the disconnection I^{uv} and the connection $\bar{I}^{uv}$ both occur in $\mathtt{t}$, then one of the terms E^{v}_{D}, $\bar{E}^{v}_{D}$, E^{va} or $\bar{E}^{va}$ occurs in $\mathtt{t}$.

Proposition 6.12. *In* **Disc***, any term is equal to a term in normal form.*

Proof. By Propositions 6.8 and 6.10, every term is equal to a term in an $ICER$-form: let us fix such a term

$$\mathtt{t} = \mathtt{I}; \mathtt{C}; \mathtt{E}^{<0}; \mathtt{E}^{\geq 0}; \bar{\mathtt{E}}^{\geq 0}; \bar{\mathtt{E}}^{<0}; \bar{\mathtt{C}}; \bar{\mathtt{I}}; \mathtt{A}; \mathtt{B}; \mathtt{S}.$$

Conditions (1) and (2) are obtained by absorbing the "excess" S-terms into other terms using equations (6), (19), (21) and (35). Conditions (3) and (4) are obtained by treating all the vertex names in $D^{add}_{\mathtt{t}}$ as "dummy" names, which are removed either by a connection or a renaming term.

For (5), suppose that $\bar{d}^{U}_{D[a]} : A \to B$ and $R^{z \mapsto a}$ both occur, and moreover A is in the domain of $\bar{d}^{U}_{D[z/a]}$. We commute the renaming term to the left to obtain $\bar{d}^{U}_{D[a]}; R^{z \mapsto a}$. But this is equal to $\bar{d}^{U}_{D[z/a]}; R^{a \mapsto a}$ by equations (38) and (39), which gets rid of the renaming term by $R^{a \mapsto a} \equiv S^a$.

For (6) and (7), we consider the three cases separately.

Case 1: E^{uv} and $\bar{E}^{uw}$ occur in $\mathtt{t}$. We commute the terms so that they occur one after the other $E^{uv}; \bar{E}^{uw}$, which by equations (14) and (16) is equal to some combination of S- and R-terms. Noticing that the rewriting procedure to obtain an ICE-form either commutes or absorbs S- and R-terms (specifically, the results from Lemma Appendix A.9 onwards apply), we conclude that $\mathtt{t}$ has an ICE-form without E^{uv} and $\bar{E}^{uw}$.

Case 2: E^u_{ab} and $\bar{E}^u_{cd}$ occur in $\mathtt{t}$. In combination with Case 1, it follows that the terms E^{uv} and $\bar{E}^{uw}$ do not occur, so that there are no obstructions for commuting E^u_{ab} next to $\bar{E}^u_{cd}$, obtaining $E^u_{ab}; \bar{E}^u_{cd}$. By equations (11), (12) and (13), this is equal to some combination of S- and R-terms, which we commute to the right as in Case 1.
Case 3: C^{uv}_{ab} and either $\bar{C}^{uv}_{cd}$ or $\bar{C}^{vu}_{cd}$ occur in $\mathtt{t}$. The latter case simply reduces to the former by equation (22). Thus suppose that $\bar{C}^{uv}_{cd}$ occurs. First, we use equation (30) to commute C^{uv}_{ab} to the right past all the $E^{<0}$-terms, and $\bar{C}^{uv}_{cd}$ to the left past all the $\bar{E}^{<0}$-terms. Next, we commute any terms of the form E^{ui} and E^{vj} to the right past the $\bar{E}^{\geq 0}$-terms and $\bar{C}^{uv}_{cd}$ using the fact that by Case 1 the terms $\bar{E}^{ui}$ and $\bar{E}^{vj}$ do not occur, together with the equations (17) and (32). Now there are no obstructions for commuting C^{uv}_{ab} to the right past all the $E^{\geq 0}$- and $\bar{E}^{\geq 0}$-terms, obtaining $C^{uv}_{ab}; \bar{C}^{uv}_{cd}$. As in Case 2, equations (11), (12) and (13) yield that this is equal to some combination of S- and R-terms, which we commute to the right as in Case 1. Finally, we return the terms of the form E^{ui} and E^{vj} back to the left past all the $\bar{E}^{\geq 0}$-terms.

For (8), suppose that both I^{uv} and $\bar{I}^{uv}$ occur in $\mathtt{t}$ such that no E- or $\bar{E}$ term containing u occurs. It follows that no E- or $\bar{E}$ term containing v occurs either: application of $\bar{I}^{uv}$ requires for u and v to have equal and opposite charge, so if the charge of u is unchanged, so is the charge of v; moreover, by (6) and (7), we may assume that no change can be reversed, so that we indeed cannot have any E- or $\bar{E}$ term containing v. But now there are no obstructions for commuting I^{uv} all the way to the right until we obtain $I^{uv}; \bar{I}^{uv}$, which is equal to $S^u; S^v$ by (14). □

Definition 6.13 (Normal form equivalence)**.** Let

$$\mathtt{t} = \mathtt{I}; \mathtt{C}; \mathtt{E}^{<0}; \mathtt{E}^{\geq 0}; \bar{\mathtt{E}}^{\geq 0}; \bar{\mathtt{E}}^{<0}; \bar{\mathtt{C}}; \bar{\mathtt{I}}; \mathtt{A}; \mathtt{B}; \mathtt{S}$$

be a term in a normal form. Define the following syntactic manipulations of $\mathtt{t}$:

1. commuting the terms inside each of the named sequences in the normal form,
2. permuting vertex names in C-terms: if the term C^{uv}_{ab} occurs, we may substitute it with C^{vu}_{ba},
3. if $d \in \{C, E, \bar{C}, \bar{E}\}$ such that $d^U_{ab}; d^U_{cd}$ occurs, we may substitute $d^U_{ab}; d^U_{cd} \mapsto d^U_{ad}; d^U_{cb}$,
4. renaming of vertices that are introduced and removed: if $a \in D^{add}_{\mathtt{t}} \cup C_{\mathtt{t}}$ and $z \in \mathbf{VN}$ does not occur in $\mathtt{t}$ or its domain, then we may substitute $\mathtt{t} \mapsto \mathtt{t}[z/a]$,
5. exchanging vertex names between renaming terms: if both $R^{a \mapsto b}$ and $R^{c \mapsto d}$ occur in $\mathtt{A}$ such that $\mathtt{N}(a) = \mathtt{N}(c)$, we may swap a and c,
6. exchanging vertex names between connections and renaming terms: if $d \in \{E, C\}$, and $\bar{d}^U_{D[a]} : A \to B$ and $R^{z \mapsto b}$ both occur such that A is in the domain of $\bar{d}^U_{D[z/a]}$, then we may substitute $\bar{d}^U_{D[a]} \mapsto \bar{d}^U_{D[z/a]}$ and $R^{z \mapsto b} \mapsto R^{a \mapsto b}$.

We say that two terms $\mathtt{t}$ and $\mathtt{s}$ in a normal form are *equivalent*, written $\mathtt{t} \sim \mathtt{s}$, if one can be obtained from the other by a sequence of the syntactic manipulations defined above.

Observing that each syntactic manipulation in Definition 6.13 is reversible, we see that $\sim$ is an equivalence relation on the set of terms in normal form.

Lemma 6.14. *Let* $\mathtt{t}$ *and* $\mathtt{s}$ *be terms in normal forms such that* $\mathtt{t} \sim \mathtt{s}$. *Then* $\mathtt{t} \equiv \mathtt{s}$.

Proof. This follows by noticing that every syntactic manipulation of Definition 6.13 keeps the term in a normal form, and moreover preserves the equality $\equiv$:

1. the terms may be commuted by equations (23), (2) and (18),
2. vertex names in C-terms may be permuted by (22),
3. the indices in repeated (dis)connections may be exchanged by (40),
4. if $a \in D^{add}_{\mathtt{t}}$, so that there is a disconnection $d^U_{D[a]}$, we use equation (9) to obtain $d^U_{D[a]} \equiv d^U_{D[z/a]}; R^{z \mapsto a}$ to introduce the desired fresh variable z; the renaming term can then be absorbed into the second occurrence of a, hence replacing a with z (the case when $a \in C_{\mathtt{t}}$ is similar),
5. the exchange of α-vertices with the same neighbour is obtained by equation (4):
$$R^{a \mapsto b}; R^{c \mapsto d} \equiv S^c; R^{a \mapsto b}; R^{c \mapsto d} \equiv R^{c \mapsto b}; R^{a \mapsto c}; R^{c \mapsto d} \equiv R^{c \mapsto b}; R^{a \mapsto d},$$
6. the last syntactic manipulation is obtained by equations (38) and (39).

□

7. From Disconnections to Reactions, Functorially

This section establishes a tight link between the category of disconnection rules (Definition 6.5) and the category of reactions (Definition 5.7) by constructing a functor $R : \mathbf{Disc} \to \mathbf{React}$, establishing soundness of the disconnection rules with respect to the reactions. Moreover, we prove that R is faithful and full up to isomorphism: a fact that entails completeness and universality (Theorems 7.3 and 7.4). In combination, the results of this section allow for algebraic reasoning about the reactions using the equations for the disconnection rules (Figure 7).

We define a function R from terms to morphisms in **React** as follows. Given a term $\mathtt{t} : A \to B$, the morphism $R(\mathtt{t}) : A \to B$ has the form

$$R(R_1(\mathtt{t}), R_2(\mathtt{t}), \mathrm{id}, \mathrm{id}),$$

where $\mathrm{id} : \mathtt{Chem}(R_1(\mathtt{t})) \to \mathtt{Chem}(R_2(\mathtt{t}))$ and $\mathrm{id} : V_A \setminus R_1(\mathtt{t}) \to V_B \setminus R_2(\mathtt{t})$ are both identity maps. Since all the terms are mapped to morphisms whose bijection and isomorphism parts are the identities, we omit these, and simply

write $R(\mathtt{t}) = (R_1(\mathtt{t}), R_2(\mathtt{t}))$. The recursive definition of this mapping is given below:

$$\begin{aligned}
R(\mathrm{id}_A) &\coloneqq (\varnothing, \varnothing) & R(E^{uv}) &\coloneqq (\{u,v\},\{u,v\}) \\
R(S^u) &\coloneqq (\{u\},\{u\}) & R(I^{uv}) &\coloneqq (\{u,v\},\{u,v\}) \\
R(R^{u\mapsto v}) &\coloneqq (\{u\},\{v\}) & R(C^{uv}_{ab}) &\coloneqq (\{u,v\},\{u,v,a,b\}) \\
R(E^u_{ab}) &\coloneqq (\{u\},\{u,a,b\}) & R(\bar{d}^{uv}_{ab}) &\coloneqq \overline{R(d^{uv}_{ab})} \\
& & R(\mathtt{t};\mathtt{s}) &\coloneqq R(\mathtt{t}); R(\mathtt{s}).
\end{aligned}$$

Observe that for all the disconnections we have $R(d^U_D) = (U, U \cup D)$.

Soundness of disconnection rules with respect to reactions is expressed as functoriality:

Proposition 7.1. *The assignment $R : \mathbf{Disc} \to \mathbf{React}$ is a dagger functor.*

Proof. Functoriality and preservation of dagger structure follow immediately from the definition. We have to show that R preserves the equalities in **Disc**, generated by the identities in Figure 7. Most of these follow immediately by assuming that the expressions on both sides of the equality have the same type, and showing that they are mapped to the same pair of sets by R. Hence we only give the cases that are less obvious or require more computation. To further simplify the notation, we omit the curly brackets of set-builder notation as well as the commas separating vertex names from each other: so e.g. $(uv, uvab)$ stands for $(\{u,v\},\{u,v,a,b\})$.

For (7), suppose that $R^{u\mapsto v}; d^U_D \equiv d^U_D; R^{u\mapsto v}$. In particular, it follows that $u, v \notin U \cup D$. Let us write $R(d^U_D) = (R_1, R_2)$, so that $u, v \notin R_1, R_2$. We use this to show that both sides of the equality evaluate to the same map:

$$(u,v);(R_1,R_2) = (\{u\} \cup R_1, R_2 \cup \{v\}) = (R_1,R_2);(u,v).$$

For (10), suppose that $d^{U'}_{ij}; \bar{h}^U_{ab}; R^{i\mapsto c}; R^{j\mapsto d} \equiv \bar{h}^U_{ab}; d^{U'}_{cd}$. Denote $R(d^{U'}_{ij}) = (u'v', u'v'ij)$, $R(d^{U'}_{cd}) = (u'v', u'v'cd)$ and $R(\bar{h}^U_{ab}) = (uvab, uv)$. Note that d and h are not $E^{\geq 0}$-terms, whence it follows that $i, j \notin U$ and $a, b \notin U'$. From the fact that the left-hand side is defined, we obtain that $\{i,j\}$ and $\{a,b\}$ are disjoint. The left-hand side is thus translated to

$$\begin{aligned}
(u'v', u'v'ij);(uvab,uv);(i,c);(j,d) &= (u'v'uvab, uvu'v'ij);(i,c);(j,d) \\
&= (u'v'uvab, cuvu'v'j);(j,d) \\
&= (u'v'uvab, dcuvu'v') \\
&= (uvu'v'ab, uvu'v'cd) \\
&= (uvab,uv);(u'v', u'v'cd),
\end{aligned}$$

which we recognise as the translation of the right-hand side.

For (16), suppose $E^{ua}; \bar{E}^{ub} \equiv S^u; R^{a\mapsto z}; R^{b\mapsto a}; R^{z\mapsto b}$. We start from the translation of the right-hand side:

$$\begin{aligned}
(u,u);(a,z);(b,a);(z,b) &= (ua,zu);(bz,ba) \\
&= (uab, bau) \\
&= (uab, uba) \\
&= (ua,ua);(ub,ub),
\end{aligned}$$

which we recognise as the translation of the left-hand side.

For (23), write $R(d^U_D) = (U, U \cup D)$ and $R(d^{U'}_{D'}) = (U', U' \cup D')$, so that we get

$$R(d^U_D; d^{U'}_{D'}) = (U \cup U', U \cup U' \cup D \cup D') = R(d^{U'}_{D'}; d^U_D).$$

□

Recall the syntactic manipulations of terms in normal form we introduced in Definition 6.13. We have seen that these manipulations preserve equality (Lemma 6.14). The following lemma is the core of the completeness argument.

Lemma 7.2. *Let* $\mathtt{t}$ *and* $\mathtt{s}$ *be terms in a normal form such that* $R(\mathtt{t}) = R(\mathtt{s})$. *Then* $\mathtt{t} \sim \mathtt{s}$.

Proof. Let us write

$$\mathtt{t} = \mathtt{I_t}; \mathtt{C_t}; \mathtt{E}^{<0}_{\mathtt{t}}; \mathtt{E}^{\geq 0}_{\mathtt{t}}; \bar{\mathtt{E}}^{\geq 0}_{\mathtt{t}}; \bar{\mathtt{E}}^{<0}_{\mathtt{t}}; \bar{\mathtt{C}}_{\mathtt{t}}; \bar{\mathtt{I}}_{\mathtt{t}}; \mathtt{A_t}; \mathtt{B_t}; \mathtt{S_t},$$
$$\mathtt{s} = \mathtt{I_s}; \mathtt{C_s}; \mathtt{E}^{<0}_{\mathtt{s}}; \mathtt{E}^{\geq 0}_{\mathtt{s}}; \bar{\mathtt{E}}^{\geq 0}_{\mathtt{s}}; \bar{\mathtt{E}}^{<0}_{\mathtt{s}}; \bar{\mathtt{C}}_{\mathtt{s}}; \bar{\mathtt{I}}_{\mathtt{s}}; \mathtt{A_s}; \mathtt{B_s}; \mathtt{S_s}.$$

Similarly, let us denote the vertex name sets in the respective renaming forms by $A_{\mathtt{t}}, B_{\mathtt{t}}, C_{\mathtt{t}}, D_{\mathtt{t}}$ and $A_{\mathtt{s}}, B_{\mathtt{s}}, C_{\mathtt{s}}, D_{\mathtt{s}}$. Let us denote the morphism $R(\mathtt{t}) = R(\mathtt{s})$ by $(R_1, R_2) : A \to B$.

First, we observe that if $E^{ua} \in \mathtt{E}^{\geq 0}_{\mathtt{t}}$, then condition (7) of normal form (Definition 6.11) implies that the charge of u cannot be increased, whence there is a vertex name $b \in \mathbf{VN}$ such that $E^{ub} \in \mathtt{E}^{\geq 0}_{\mathtt{s}}$. Similarly, by condition (6) of normal form, if $E^u_{ab} \in \mathtt{E}^{<0}_{\mathtt{t}}$, then $E^u_{cd} \in \mathtt{E}^{<0}_{\mathtt{s}}$ for some $c, d \in \mathbf{VN}$; and if $C^{uv}_{ab} \in \mathtt{C_t}$, then $C^{uv}_{cd} \in \mathtt{C_s}$ for some $c, d \in \mathbf{VN}$. By condition (8) of normal form, if $I^{uv} \in \mathtt{I_t}$ then $I^{uv} \in \mathtt{I_s}$. Since the connections cannot be undoing the disconnections, a similar inclusion up to α-vertices holds for them. Thus we obtain that the sequences of disconnections and connections must coincide, up to renaming the α-vertices.

Next, suppose that $R^{a \mapsto b} \in \mathtt{A_t}$, so that $a \in A_{\mathtt{t}}$ and $b \in B_{\mathtt{t}}$. There are four cases.

Case 1: $a \notin D^{add}_{\mathtt{t}}$ and $b \notin C_{\mathtt{t}}$. This implies that $a \in R_1$ and $b \in R_2$. Moreover, if $b \in R_1$, then condition (5) of normal form yields that $\mathtt{N}(a) \neq \mathtt{N}(b)$. It follows that either $R^{a \mapsto b} \in \mathtt{A_s}$, or both $\bar{d}^U_{D[a]}$ and $R^{z \mapsto b}$ occur in $\mathtt{s}$ such that $\bar{d}^U_{D[z/a]}$ is defined. But in the latter case the vertex names a and z may be exchanged by syntactic manipulation 6 (Definition 6.13), so that we may assume $R^{a \mapsto b} \in \mathtt{A_s}$.

Case 2: $a \notin D^{add}_{\mathtt{t}}$ and $b \in C_{\mathtt{t}}$. This means that $R^{b \mapsto d} \in \mathtt{B_t}$ for some $d \in D_{\mathtt{t}}$, and for some $x \in \mathbf{VN}$, we have $R^{d \mapsto x} \in \mathtt{A_t}$. Condition (3) of normal form implies that $d \notin D^{add}_{\mathtt{t}}$, so that we have $a \in R_1$ and $d \in R_1 \cap R_2$. If $x \notin C_{\mathtt{t}}$, then by Case 1, $R^{d \mapsto x} \in \mathtt{A_s}$, so that also $R^{a \mapsto z} \in \mathtt{A_s}$ and $R^{z \mapsto d} \in \mathtt{B_s}$ for some $z \in \mathbf{VN}$. If $x \in C_{\mathtt{t}}$, then we inductively repeat Case 2. By syntactic manipulation 4, we may assume that $R^{a \mapsto b} \in \mathtt{A_s}$ and $R^{b \mapsto d} \in \mathtt{B_s}$.

Case 3: $a \in D^{add}_{\mathtt{t}}$ and $b \notin C_{\mathtt{t}}$. Thus there is a disconnection $d^U_{D[a]} \in \mathtt{t}$, so that $d^U_{D[x/a]} \in \mathtt{s}$ for some $x \in \mathbf{VN}$. This implies $a \notin R_1 \cup R_2$ and $b \in R_2$. As in Case 1, it follows that $R^{z \mapsto b} \in \mathtt{A_s}$ for some $z \in \mathbf{VN}$. Moreover, in this case we must have $\mathtt{N}(x) = \mathtt{N}(z)$, whence by syntactic manipulation 5 we may assume that $R^{x \mapsto b} \in \mathtt{A_s}$ with $x \in D^{add}_{\mathtt{s}}$. By syntactic manipulation 4, we may assume that $R^{a \mapsto b} \in \mathtt{A_s}$ and $d^U_{D[a]} \in \mathtt{s}$.

Case 4: $a \in D^{add}_{\mathtt{t}}$ and $b \in C_{\mathtt{t}}$. We thus have $a, b \notin R_1 \cup R_2$. This means that $R^{b \mapsto d} \in \mathtt{B_t}$ for some $d \in D_{\mathtt{t}}$, so that $R^{d \mapsto x} \in \mathtt{A_t}$ for some $x \in \mathbf{VN}$. Condition (3) of normal form implies that $d \notin D^{add}_{\mathtt{t}}$, so that $d \in R_1 \cap R_2$ and either Case 1 or Case 2 applies to $R^{d \mapsto x}$. In both cases we conclude that $R^{d \mapsto x} \in \mathtt{A_s}$. Since $d \in R_2$, we have $R^{w \mapsto d} \in \mathtt{B_s}$ for some $w \in \mathbf{VN}$. Since there is a disconnection $d^U_{D[a]} \in \mathtt{t}$, we have $d^U_{D[y/a]} \in \mathtt{s}$ for some $y \in \mathbf{VN}$. Note that we have $\mathtt{N}(y) = \mathtt{N}(w)$. Consequently, by syntactic manipulation 5, we may assume that $R^{y \mapsto w} \in \mathtt{A_s}$ with $y \in D^{add}_{\mathtt{s}}$ and $w \in C_{\mathtt{s}}$. By syntactic manipulation 4, we may assume that $R^{a \mapsto b} \in \mathtt{A_s}$, $R^{b \mapsto d} \in \mathtt{B_s}$ and $d^U_{D[a]} \in \mathtt{s}$.

Thus we have shown that $\mathtt{t}$ and $\mathtt{s}$ have the same renaming sequences (up to $\sim$), and up to the syntactic manipulations, $D^{add}_{\mathtt{t}} = D^{add}_{\mathtt{s}}$ and $D^{remove}_{\mathtt{t}} = D^{remove}_{\mathtt{s}}$.

If $S^u \in \mathtt{S_t}$, then $u \in R_1 \cap R_2$ and, by conditions (1) and (2) of normal form, u does not occur anywhere else in $\mathtt{t}$. The argument so far entails that $u \notin U_{\mathtt{s}} \cup A_{\mathtt{s}} \cup B_{\mathtt{s}}$, so that $S^u \in \mathtt{S_s}$. Thus $\mathtt{S_t} = \mathtt{S_s}$.

Now the only difference left between $\mathtt{t}$ and $\mathtt{s}$ is in which order the vertex names are introduced and removed. This is taken care of precisely by syntactic manipulations 1 and 3. □

Combining the above lemma with the results from the previous section, we conclude that the functor $R : \mathbf{Disc} \to \mathbf{React}$ is faithful. We spell this out in detail in the following:

Theorem 7.3 (Completeness)**.** *For all terms* $\mathtt{t}$ *and* $\mathtt{s}$*, we have* $\mathtt{t} \equiv \mathtt{s}$ *in* **Disc** *if and only if* $R(\mathtt{t}) = R(\mathtt{s})$ *in* **React***.*

Proof. The 'only if' direction is functoriality (Proposition 7.1). The 'if' direction follows from the fact that every term is equal to a term in normal form (Proposition 6.12) and from Lemmas 7.2 and 6.14. □

The argument for universality turns out to be much simpler than that for completeness. However, in combination with Theorem 7.3, it gives a rather strong representation result for reactions: not only can every reaction be decomposed into a sequence of disconnection rules, but this sequence is also unique, up to changing the vertex names and up to the equations in **Disc**. In abstract terms, the statement of universality is that the functor $R : \mathbf{Disc} \to \mathbf{React}$ is full up to isomorphism in **React**. As for completeness, we spell out the details:

Theorem 7.4 (Universality)**.** *Given a reaction* $r : A \to C$ *in* **React***, there is a term* $\mathtt{t} : A \to B$ *in* **Disc** *and an isomorphism* $\iota : B \xrightarrow{\sim} C$ *in* **React** *such that* $R(\mathtt{t}); \iota = r$*.*

Proof. Observe that every reaction $r : A \to C$ factorises as

$$(U_A, U_B, \mathrm{id}, \mathrm{id}); (\varnothing, \varnothing, !, \iota),$$

where $(U_A, U_B) : A \to B$ is some reaction and $\iota : B \to C$ is an isomorphism of labelled graphs. Now, we may disconnect all possible bonds inside U_A, and then connect all possible bonds to obtain U_B. The fact that U_A and U_B have the same atom vertices and the same net charge guarantee that this can always

be done. Precisely, the sought-after term $\mathtt{t} : A \to B$ is then given by

$$\prod_{\substack{u\in\mathtt{Crg}^+(U_A)\\ v\in\mathtt{Crg}^-(U_A)}} (I^{uv})^{\mathtt{ion}(m_A(u,v))}; \prod_{u,v\in\mathtt{Chem}(U_A)} \prod_{i=1}^{\mathtt{cov}(m_A(u,v))} C^{uv}_{a_ib_i};$$

$$\prod_{u\in\mathtt{Crg}^-(U_A)} \prod_{i=1}^{-\tau_A^{\mathtt{Crg}}(u)} E^u_{a_ib_i}; \prod_{u\in\mathtt{Chem}(U_A)} \prod_{i=1}^{\mathbf{v}\tau_A^{\mathrm{At}}(u)-\max\left(\tau_A^{\mathtt{Crg}}(u),0\right)} E^{ua_i};$$

$$\prod_{u\in\mathtt{Chem}(U_B)} \prod_{i=1}^{\mathbf{v}\tau_B^{\mathrm{At}}(u)-\max\left(\tau_B^{\mathtt{Crg}}(u),0\right)} \bar{E}^{ua_i}; \prod_{u\in\mathtt{Crg}^-(U_B)} \prod_{i=1}^{-\tau_B^{\mathtt{Crg}}(u)} \bar{E}^u_{a_ib_i};$$

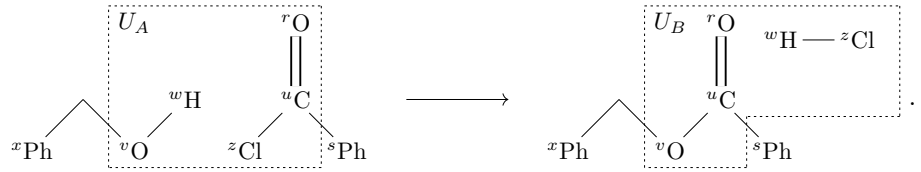
$$\prod_{u,v\in\mathtt{Chem}(U_B)} \prod_{i=1}^{\mathtt{cov}(m_B(u,v))} \bar{C}^{uv}_{a_ib_i}; \prod_{\substack{u\in\mathtt{Crg}^+(U_B)\\ v\in\mathtt{Crg}^-(U_B)}} \left(\bar{I}^{uv}\right)^{\mathtt{ion}(m_B(u,v))};$$

$$\prod_{a\in\alpha(U_A)\setminus D} R^{a\mapsto b_a}; \prod_{b\in\alpha(U_B)} R^{a_b\mapsto b}; \prod_{u\in U_B} S^u,$$

where the vertex names introduced by the C- and $E^{<0}$-terms are chosen so that they do not appear anywhere in A or B, and their set is denoted by I. The vertex names removed by the $\bar{C}$- and $\bar{E}^{<}$-terms are chosen from $U_A \cup I$ such that the connection is well-typed: their set is denoted by D. Similarly, the α-vertices appearing in the $E^{\geq 0}$- and $\bar{E}^{\geq 0}$-terms are chosen from $U_A \cup I$ such that the terms are well-typed. The vertex names introduced by the $R^{a\mapsto b_a}$-terms, where $a \in \alpha(U_A) \setminus D$, are chosen so that they do not appear in A, B or I: their set is denoted by R. Finally, the vertex names removed by the $R^{a_b\mapsto b}$-terms, where $b \in \alpha(U_B)$ are chosen from $I \cup R$ in such a way that the terms are well-typed.

Note that while the term we obtain is in an ICE-form, it will not, in general, be in normal form. □

Example 7.5. Consider the reaction from Figure 2 (formation of benzyl benzoate from benzoyl chloride and benzyl alcohol), which we redraw with vertex names below. Here both b and i are identity maps, and Ph stands for the phenyl group:



Following the procedure of Theorem 7.4, the reaction decomposes into the following sequence of (dis)connection rules:

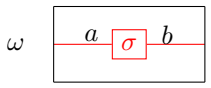
$$C^{zu}_{ab}; C^{vw}_{cd}; C^{ru}_{ij}; C^{ru}_{nm}; E^{vc}; E^{wd}; E^{za}; E^{ub}; E^{ri}; E^{uj}; E^{rn}; E^{um};$$
$$\bar{E}^{vc}; \bar{E}^{wd}; \bar{E}^{za}; \bar{E}^{ub}; \bar{E}^{ri}; \bar{E}^{uj}; \bar{E}^{rn}; \bar{E}^{um}; \bar{C}^{ru}_{ij}; \bar{C}^{ru}_{nm}; \bar{C}^{wz}_{da}; \bar{C}^{uv}_{bc};$$
$$S^z; S^u; S^v; S^w; S^r.$$

The normal form of the above sequence is given by:

$$C^{zu}_{ab}; C^{vw}_{cd}; \bar{C}^{wz}_{da}; \bar{C}^{uv}_{bc}; S^r.$$

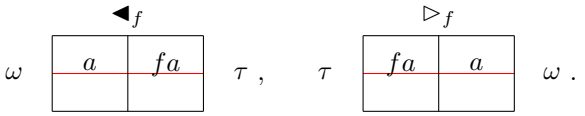
8. Layered Props

Layered props were introduced in [15] as categorical models for diagrammatic reasoning about systems with several levels of description. They have been employed to account for partial explanations and semantic analysis in the context of electrical circuit theory, chemistry, and concurrency. Formally, a layered prop is essentially a functor $\Omega : P \to \mathbf{StrMon}$ from a poset P to the category of strict monoidal categories, together with a right adjoint for each monoidal functor in the image of Ω. Given $\omega \in P$, we denote a morphism $\sigma : a \to b$ in $\Omega(\omega)$ by the box as follows:



We think of σ as a *process* with an input a and an output b happening in the *context* ω. Note, however, that these diagrams are not merely a convenient piece of notation that capture our intuition: they are a completely formal syntax of string diagrams, describing morphisms in a certain subcategory of pointed profunctors [15].

The monoidal categories in the image of Ω are thought of as languages describing the same system at different levels of granularity, and the functors are seen as translations between the languages. Given $\omega \leq \tau$ in P, let us write $f \coloneqq \Omega(\omega \leq \tau)$. Then, for each $a \in \Omega(\omega)$ we have the following morphisms:



The reason for having morphisms in both directions is that we want to be able to "undo" the action of a translation while preserving a linear reasoning flow. The two morphisms will not, in general, be inverse to each other: rather, they form an adjoint pair. This corresponds to the intuition that some information is gained by performing the translation, and that the translation in the reverse direction is our best guess, or an approximation, not a one-to-one correspondence.

There are two ways to compose morphisms in parallel in a layered prop: internally within a monoidal category $\Omega(\omega)$ using its own monoidal product (composition inside a context), and externally using the Cartesian monoidal structure of **StrMon** (doing several processes in different contexts in parallel). We represent the latter by stacking the boxes on top of each other. Additional morphisms of a layered prop ensure that the internal and the external monoidal structures interact in a coherent way. Finally, a layered prop comes with "deduction rules" (2-cells) which allow transforming one process into another one. We refer the reader to [15] for the details.

In this work, the processes in context will be the retrosynthetic disconnection rules (Section 6) and the chemical reactions (Section 5). The context describes the reaction environment as well as the level of granularity at which the synthesis is happening (i.e. what kinds of disconnection rules are available). The objects in the monoidal categories are given by molecular entities and their parts: this is the subject of the next section.

9. Retrosynthesis in Layered Props

This penultimate section puts to use all three perspectives on chemical processes developed so far – reaction schemes, reactions, and disconnection rules. We combine these with layered props (Section 8) to propose a mathematical framework for retrosynthesis (Section 2). The layers of the layered prop we propose as the habitat for retrosynthesis all share the same set of objects – namely, the chemical graphs. The morphisms of a layer are either matchings, disconnection rules or reactions, parameterised by environmental molecules (these can act as solvents, reagents or catalysts).

Given a finite set M of molecular entities, let us enumerate the molecular entities in M as $M_1, \ldots, M_k$. Given a list natural numbers $n = (n_1, \ldots, n_k)$, we denote by $n_1M_1 + \cdots + n_kM_k$ the molecular graph obtained by taking the disjoint union of n_i copies of M_i for all $i = 1, \ldots, k$. We define three classes of symmetric monoidal categories parameterised by finite sets of molecular entities as follows.

Definition 9.1. Let M be a finite set of molecular entities. We define the categories M-**Match**, M-**React** and M-**Disc** as having the chemical graphs as objects. The morphisms are defined as follows:

- in M-**Match**, a morphism $A \xrightarrow{m,r} B$ is given by a matching $m : A \to B$ together with an injection $r : r_1M_1 + \cdots + r_kM_k \to B$ preserving the atom labels such that $\mathrm{im}(m) \cup \mathrm{im}(r) = B$, and $\mathrm{im}(m) \cap \mathrm{im}(b) = m(\alpha(A))$. The composition of $A \xrightarrow{m,r} B \xrightarrow{n,s} C$ is given by

$$m; n : A \to C \text{ and } r; n + s : (r_1 + s_1)M_1 + \cdots + (r_k + s_k)M_k \to C.$$

- in M-**React**, a morphism $A \to B$ is a reaction $n_1M_1 + \cdots + n_kM_k + A \xrightarrow{r} B$ (i.e. a morphism in **React**). Given another reaction $m_1M_1 + \cdots + m_kM_k + B \xrightarrow{s} C$, the composite $A \to C$ is given by

$$(r + \mathrm{id}_{m_1M_1 + \cdots + m_kM_k}); s : (n_1 + m_1)M_1 + \cdots + (n_k + m_k)M_k + A \to C.$$

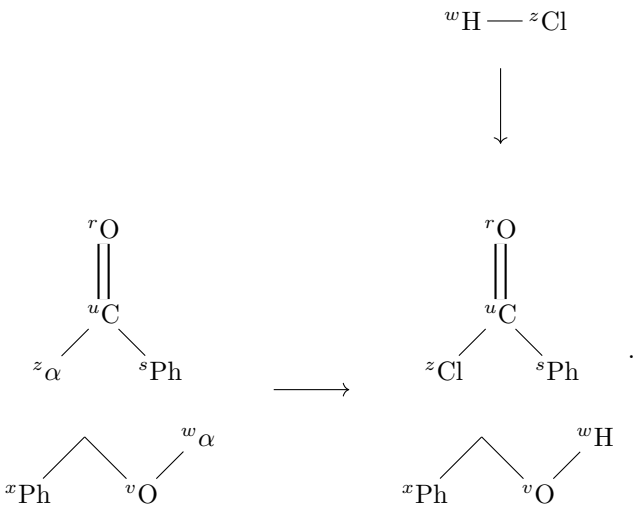
- in M-**Disc**, a morphism $A \to B$ is given by a morphism $n_1M_1 + \cdots + n_kM_k + A \xrightarrow{d} B$ in **Disc**. Given another morphism $m_1M_1 + \cdots + m_kM_k + B \xrightarrow{h} C$ the composite $A \to C$ is given by

$$d; h : (n_1 + m_1)M_1 + \cdots + (n_k + m_k)M_k + A \to C.$$

If $M = \varnothing$, we may omit the prefix.

The idea is that the set M models the reaction environment: the parametric definitions above capture the intuition that there is an unbounded supply of these molecules in the environment. The categories M-**React** and M-**Disc** are the parameterised [41] versions of **React** and **Disc**: a morphism $A \to B$ implicitly has a finite number of copies of molecules from M in its domain. A morphism $A \to B$ in M-**Match** may be seen as a reaction which preserves the structure of A as it is, while potentially breaking up and rearranging the molecular entities in M. We proceed to give an example of this.

Example 9.2. A morphism in M-**Match** is a matching such that the environment contains enough "building material" to cover the complement of the image of the matching. We give an example below, taking $M = \{\texttt{HCl}\}$, where the horizontal map is the matching, and the vertical map is the injection (cf. Figure 2):



We formalise the fact that morphisms in M-**Match** look like special cases of reactions by noticing that for every M there is an identity-on-objects functor

$$M\text{-}I : M\text{-}\mathbf{Match} \to M\text{-}\mathbf{React}$$

defined by $(m, r) \mapsto m|_{\texttt{Chem}(A)} + r$, where A is the domain of (m, r). Thus we have the following situation, for every finite set of molecular entities M:

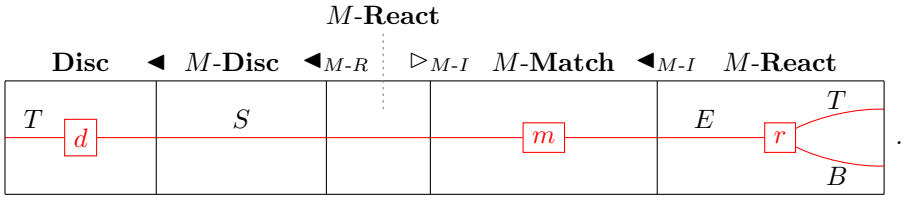
$$M\text{-}\mathbf{Match} \xrightarrow{M\text{-}I} M\text{-}\mathbf{React} \xleftarrow{M\text{-}R} M\text{-}\mathbf{Disc} \quad , \tag{41}$$

where M-R is defined by the action of the functor R constructed in Section 7. Additionally, for every pair of finite sets of molecular entities such that $M \subseteq N$, there is an inclusion functor for each of the three classes of categories.

Definition 9.3 (Retrosynthetic step)**.** A *retrosynthetic step* consists of

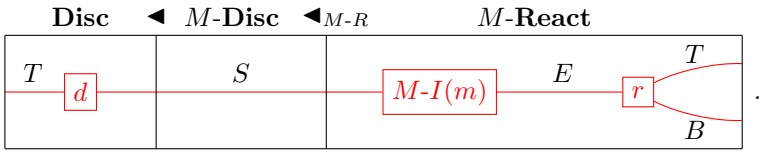
- molecular graphs T and B, called the *target*, and the *byproduct*,
- a finite set of molecular entities M, called the *environment*,
- a chemical graph S, whose connected components are called the *synthons*,
- a molecular graph E, whose connected components are called the *synthetic equivalents*,
- morphisms $d \in \mathbf{Disc}(T, S)$, $m \in M\text{-}\mathbf{Match}(S, E)$, and $r \in M\text{-}\mathbf{React}(E, T + B)$.

Proposition 9.4. *The data of a retrosynthetic step are equivalent to existence of the following morphism (1-cell) in the layered prop generated by the diagram* (41)*:*

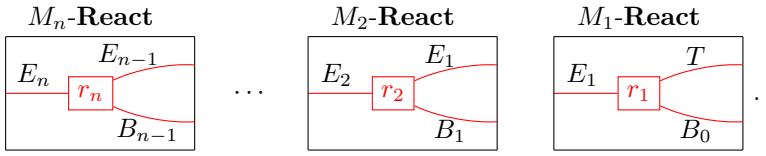


The morphism in the above proposition should be compared to the informal diagram in Figure 2. The immediate advantage of presenting a retrosynthetic step as a morphism in a layered prop is that it illustrates how the different parts of the definition fit together in a highly procedural manner. Equally importantly, this presentation is fully compositional: note that the three morphisms constituting a retrosynthetic step can be divided between several parties (e.g. different labs or computers), so long as their boundaries match in the specified way. Moreover, one can reason about different components of the step while preserving a precise mathematical interpretation, so long as one sticks to the rewrites (2-cells) of the layered prop: we illustrate this in the following proposition.

Proposition 9.5. *There is a rewrite (2-cell) from the 1-cell of Proposition 9.4 to the following 1-cell:*



Definition 9.6 (Retrosynthetic sequence)**.** A *retrosynthetic sequence* for a target molecular entity T is a sequence of morphisms $r_1 \in M_1\text{-}\mathbf{React}(E_1, T + B_0)$, $r_2 \in M_2\text{-}\mathbf{React}(E_2, E_1 + B_1)$, $\ldots$, $r_n \in M_1\text{-}\mathbf{React}(E_n, E_{n-1} + B_{n-1})$ such that the codomain of r_{i+1} is the disjoint union of the domain of r_i with some other molecular graph:



Thus a retrosynthetic sequence is a chain of reactions, together with reaction environments, such that the products of one reaction can be used as the reactants for the next one, so that the reactions can occur one after another (assuming that the products can be extracted from the reaction environment, or one environment transformed into another one). In the formulation of a generic retrosynthesis procedure below, we shall additionally require that each reaction in the sequence comes from "erasing" everything but the rightmost cell in a retrosynthetic step.

We are now ready to formulate step-by-step retrosynthetic analysis. The procedure is a high-level mathematical description that, we suggest, is flexible enough to capture all instances of retrosynthetic algorithms. As a consequence, it can have various computational implementations. Let T be some fixed molecular entity. We initialise by setting $i = 0$ and $E_0 \coloneqq T$.

1. Choose a subset $\mathcal{D}$ of sequences of disconnection rules,
2. Provide at least one of the following:
   (a) a finite set of reaction schemes $\mathcal{S}$,
   (b) a function $\mathfrak{F}$ from molecular graphs to finite sets of molecular graphs,
3. Search for a retrosynthetic step with morphisms $d \in \mathbf{Disc}(E_i, S)$, $m \in M\text{-}\mathbf{Match}(S, E)$, and $r \in M\text{-}\mathbf{React}(E, E_i + B_i)$ such that $d \in \mathcal{D}$, and we have at least one of the following:
   (a) there is an $s \in \mathcal{S}$ such that the reaction r is an instance of s,
   (b) $E_i + B_i \in \mathfrak{F}(E)$;

   if successful, set $E_{i+1} \coloneqq E$, $M_{i+1} \coloneqq M$, $r_{i+1} \coloneqq r$ and proceed to Step 4; if unsuccessful, stop,
4. Check if the molecular entities in E_{i+1} are known (commercially available): if yes, terminate; if no, increment $i \mapsto i + 1$ and return to Step 1.

Note how our framework is able to incorporate both template-based and template-free retrosynthesis, corresponding to the choices between (a) and (b) in Step 2: the set $\mathcal{S}$ is the template, while the function $\mathfrak{F}$ can be a previously trained algorithm, or other unstructured empirical model of reactions. We can also consider hybrid models by providing both $\mathcal{S}$ and $\mathfrak{F}$, hence allowing for combinations of existing algorithms.

We take the output retrosynthetic sequence to always come with a specified reaction environment for each reaction. Currently existing tools rarely provide this information (mostly for complexity reasons), and hence, in our framework, correspond to the set M always being empty in Step 3.

The retrosynthetic steps outputted by the above procedure are highly tunable: the choice of the set $\mathcal{D}$ determines what kinds of bonds are disconnected (one could, for example, put an upper bound to the number of disconnected covalent bonds), while the set $\mathcal{S}$ can be used to enforce the presence of a functional group. Introducing negative application conditions for double pushout rewriting [42, 43] would further allow enforcing an absence of a functional group[3]. A rudimentary form of specificity is obtained by minimising the size of the byproduct molecular graph B_i. Further adjustments increasing the yield include choosing the environment M_i (which, inter alia, can function as a catalyst) as well as introducing protection-deprotection steps.

Steps 1 and 2 both require making some choices. Two approaches to reduce the number of choices, as well as the search space in Step 3, have been proposed in the automated retrosynthesis literature: to use molecular similarity [7], or machine learning [9]. Chemical similarity can be used to determine which disconnection rules, reactions and environment molecules are actually tried: e.g. in Step 1, disconnection rules that appear in syntheses of molecules similar to T can be prioritised.

[3]We thank an anonymous reviewer of *Theoretical Computer Science* for bringing up this point.

Ideally, each unsuccessful attempt to construct a retrosynthetic step in Step 3 should return some information on why the step failed: e.g. if the codomain of a reaction fails to contain E_i, then the output should be the codomain and a measure of how far it is from E_i. Similarly, if several reactions are found in Step 3, some of which result in products O that do not contain E_i, the step should suggest minimal alterations to E such that these reactions do not occur. This can be seen as a *deprotection* step: the idea is that in the next iteration the algorithm will attempt to construct (by now a fairly complicated) E, but now there is a guarantee this is worth the computational effort, as this prevents the unwanted reactions from occurring (*protection* step). Passing such information between the layers would take the full advantage of the layered prop formalism.

10. Discussion and Future Work

This article has discussed in detail three mathematical perspectives on chemical processes: reaction schemes, category of reactions and disconnection rules. Reaction schemes are a compact way to store chemical reaction data, and generate all the formal chemical reactions via double pushout rewriting. The category of reactions captures combinatorially all the theoretically possible chemical transformations of graphs, as well as providing a uniform notion of composition for reactions. Disconnection rules provide the fine grained, low level syntax of all chemically feasible local graph transformations. The completeness and universality results of Section 7 show that the disconnection rules and reactions are tightly linked, further motivating the use of disconnection rules, hitherto only appearing informally in the (computational) retrosynthesis literature, for both storing reaction data and as part of retrosynthetic analysis.

Universality can be thought of as a consistency result for reactions: their definition captures exactly those rearrangements of chemical graphs which result from local, chemically motivated rewrite rules. Completeness says that there is no redundancy in the representation: treating the (dis)connection rules as terms, the terms can be endowed with equations such that the terms describing the same reaction are identified. As the decomposition of a reaction into a sequence of (dis)connection rules is algorithmic, these results can be used to automatically break a reaction (or its part) into smaller components: the purpose can be, *inter alia*, retrosynthetic analysis or storing reaction data in a systematic way.

The main conceptual contributions of formulating retrosynthesis in layered props are the explicit mathematical descriptions of retrosynthetic steps (Definition 9.3) and sequences (Definition 9.6), which allows for a precise formulation of the entire process, as well as of more fine-grained concepts.

10.1. Future Work

Chemical Questions. While stereochemistry is relatively straightforward to account for on the level of chemical graphs and reactions (Subsection 3.1), it is unclear how to do this for the disconnection rules, as they only operate at one or two vertices at a time. A more straightforward extension of disconnection category would introduce energy and dynamics into the disconnection rules by quantifying how much energy each (dis)connection (in a particular context) requires to occur.

While in the current article we showed how to account for the available disconnection rules, reactions and environmental molecules as part of the retrosynthetic reaction search, the general formalism of layered props immediately suggests how to account for other environmental factors (e.g. temperature and pressure). Namely, these should be represented as posets which control the morphisms that are available between the chemical compounds. One idea for accounting for the available energy is via the disconnection rules: the higher the number of bonds that we are able to break in one step, the more energy is required to be present in the environment.

Computational Questions. Given the algorithmic nature of both completeness and universality proofs, the next step is to implement both. The first algorithm would take an arbitrary reaction as an input, and output a sequence of disconnection rules representing it. The second algorithm would decide whether two terms are equal or not, implementing the normalisation procedure. Another direction for connecting this work with more standard approaches to computational chemistry would be translating our formalism to a widely used notation such as SMILES [44, 45].

On the side of retrosynthetic design, the crucial next step is to take existing retrosynthesis algorithms and encode them in our framework. This requires implementing the morphisms of the layered prop in the previous section in some software. As the morphisms in a layered prop are represented by string diagrams, one approach is to use proof formalisation software specific to string diagrams and their equational reasoning, such as [46]. Alternatively, these morphisms could be encoded in a programming language like python or Julia. The latter is especially promising, as there are modules formalising category theory available for it [47, 48]. As a lower level description, the disconnection rules and the reactions presented could be encoded in some graph rewriting language, such as Kappa [49, 50, 51, 52], which is used to model systems of interacting agents, or MØD [53, 54, 55, 52], which represents molecules as labelled graphs and generating rules for chemical transformations as spans of graphs (akin to this work).

Mathematical Questions. An important mathematical development is to introduce monoidal terms into the disconnection category, so as to allow parallel reactions, making the discussion in Section 9 mathematically rigorous, as well as allowing for the usage of graphical calculi for monoidal categories [14]. Another mathematical question is whether the categories **Disc** and **React** have any interesting categorical structure, such as being restriction categories [56].

At the level of the layered prop formalism, the next step is to model translations between the reaction environments as functors of the form M-**React** $\to$ N-**React**. This would allow presenting a retrosynthetic sequence as a single, connected diagram, closely corresponding to actions to be taken in a lab. Similarly, we note that the informal algorithmic description in Section 9 could be presented internally in a layered prop: Steps 1 and 2 amount to choosing subcategories of **Disc** and **React**.

Acknowledgements

We thank the anonymous reviewers of *Theoretical Computer Science*, as well as of the conference papers (ICTAC 2023 and ICTAC 2024) for suggestions

vastly improving the presentation of the article. FZ acknowledges support from EPSRC grant EP/V002376/1, MIUR PRIN P2022HXNSC, and ARIA Safeguarded AI TA1.1 grant n.8777242.

References

[1] S. Warren, P. Wyatt, Organic synthesis: the disconnection approach, 2nd Edition, Wiley, Hoboken, N.J, 2008.

[2] F. Strieth-Kalthoff, F. Sandfort, M. H. S. Segler, F. Glorius, Machine learning the ropes: principles, applications and directions in synthetic chemistry, Chemical Society Reviews 49 (17) (2020) 6154–6168. doi:10.1039/C9CS00786E.

[3] Y. Sun, N. V. Sahinidis, Computer-aided retrosynthetic design: fundamentals, tools, and outlook, Current Opinion in Chemical Engineering 35 (2022) 100721. doi:10.1016/j.coche.2021.100721.

[4] E. J. Corey, Robert Robinson lecture. Retrosynthetic thinking – essentials and examples, Chemical society reviews 17 (1988) 111–133. doi:10.1039/CS9881700111.

[5] S. Warren, Designing organic syntheses: a programmed introduction to the synthon approach, John Wiley & Sons, 1991.

[6] J. Law, Z. Zsoldos, A. Simon, D. Reid, Y. Liu, S. Y. Khew, A. P. Johnson, S. Major, R. A. Wade, H. Y. Ando, Route designer: A retrosynthetic analysis tool utilizing automated retrosynthetic rule generation, Journal of Chemical Information and Modeling 49 (3) (2009) 593–602. doi:10.1021/ci800228y.

[7] C. W. Coley, L. Rogers, W. H. Green, K. F. Jensen, Computer-assisted retrosynthesis based on molecular similarity, ACS central science 3 (12) (2017) 1237–1245. doi:10.1021/acscentsci.7b00355.

[8] C. W. Coley, W. H. Green, K. F. Jensen, Machine learning in computer-aided synthesis planning, Accounts of chemical research 51 (5) (2018) 1281–1289. doi:10.1021/acs.accounts.8b00087.

[9] K. Lin, Y. Xu, J. Pei, L. Lai, Automatic retrosynthetic route planning using template-free models, Chemical science (Cambridge) 11 (12) (2020) 3355–3364. doi:10.1039/c9sc03666k.

[10] S. Chen, Y. Jung, Deep retrosynthetic reaction prediction using local reactivity and global attention, JACS Au 1 (10) (2021) 1612–1620.

[11] U. V. Ucak, I. Ashyrmamatov, J. Ko, J. Lee, Retrosynthetic reaction pathway prediction through neural machine translation of atomic environments, Nature communications 13 (1) (2022) 1186–1186. doi:10.1038/s41467-022-28857-w.

[12] J. Dong, M. Zhao, Y. Liu, Y. Su, X. Zeng, Deep learning in retrosynthesis planning: datasets, models and tools, Briefings in Bioinformatics 23 (1) (2022) bbab391. doi:10.1093/bib/bbab391.

[13] M. Filice, J. M. Guisan, J. M. Palomo, Recent trends in regioselective protection and deprotection of monosaccharides, Current Organic Chemistry 14 (6) (2010) 516–532. doi:10.2174/138527210790820276.

[14] R. Piedeleu, F. Zanasi, An Introduction to String Diagrams for Computer Scientists, Cambridge University Press, 2024, to appear, available at https://arxiv.org/abs/2305.08768.

[15] L. Lobski, F. Zanasi, String diagrams for layered explanations, Electronic Proceedings in Theoretical Computer Science 380 (2023) 362–382. doi:10.4204/eptcs.380.21.

[16] O. Bournez, L. Ibănescu, H. Kirchner, From chemical rules to term rewriting, Electronic Notes in Theoretical Computer Science 147 (1) (2006) 113–134, proceedings of the 6th International Workshop on Rule-Based Programming (RULE 2005). doi:10.1016/j.entcs.2005.06.040.

[17] E. Gale, L. Lobski, F. Zanasi, A categorical approach to synthetic chemistry, in: E. Ábrahám, C. Dubslaff, S. L. T. Tarifa (Eds.), Theoretical Aspects of Computing – ICTAC 2023, Springer Nature Switzerland, Cham, 2023, pp. 276–294. doi:10.1007/978-3-031-47963-2_17.

[18] E. Gale, L. Lobski, F. Zanasi, Disconnection rules are complete for chemical reactions (2024). arXiv:2410.01421.

[19] B. Fong, D. I. Spivak, An Invitation to Applied Category Theory: Seven Sketches in Compositionality, Cambridge University Press, Cambridge, 2019. doi:10.1017/9781108668804.

[20] E. J. Corey, X. Cheng, The logic of chemical synthesis, John Wiley, New York, 1989.

[21] J. Clayden, N. Greeves, S. Warren, Organic chemistry, 2nd Edition, Oxford University Press, Oxford, 2012.

[22] M. E. Fortunato, C. W. Coley, B. C. Barnes, K. F. Jensen, Data augmentation and pretraining for template-based retrosynthetic prediction in computer-aided synthesis planning, Journal of chemical information and modeling 60 (7) (2020) 3398–3407. doi:10.1021/acs.jcim.0c00403.

[23] C. Yan, P. Zhao, C. Lu, Y. Yu, J. Huang, RetroComposer: Composing templates for template-based retrosynthesis prediction, Biomolecules 12 (9) (2022) 1325. doi:10.3390/biom12091325.

[24] V. R. Somnath, C. Bunne, C. W. Coley, A. Krause, R. Barzilay, Learning graph models for retrosynthesis prediction (2021). arXiv:2006.07038.

[25] A. Matwijczuk, D. Karcz, R. Walkowiak, J. Furso, B. Gładyszewska, S. Wybraniec, A. Niewiadomy, G. P. Karwasz, M. Gagoś, Effect of solvent polarizability on the keto/enol equilibrium of selected bioactive molecules from the 1, 3, 4-thiadiazole group with a 2, 4-hydroxyphenyl function, The Journal of Physical Chemistry A 121 (7) (2017) 1402–1411. doi:10.1021/acs.jpca.6b08707.

[26] A. G. Cook, P. M. Feltman, Determination of solvent effects on keto–enol equilibria of 1, 3-dicarbonyl compounds using NMR, Journal of chemical education 84 (11) (2007) 1827. doi:10.1021/ed084p1827.

[27] G. Marcou, J. Aires de Sousa, D. A. R. S. Latino, A. de Luca, D. Horvath, V. Rietsch, A. Varnek, Expert system for predicting reaction conditions: The Michael reaction case, Journal of Chemical Information and Modeling 55 (2) (2015) 239–250. doi:10.1021/ci500698a.

[28] H. Gao, T. J. Struble, C. W. Coley, Y. Wang, W. H. Green, K. F. Jensen, Using machine learning to predict suitable conditions for organic reactions, ACS central science 4 (11) (2018) 1465–1476. doi:10.1021/acscentsci.8b00357.

[29] E. Walker, J. Kammeraad, J. Goetz, M. T. Robo, A. Tewari, P. M. Zimmerman, Learning to predict reaction conditions: Relationships between solvent, molecular structure, and catalyst, Journal of Chemical Information and Modeling 59 (9) (2019) 3645–3654. doi:10.1021/acs.jcim.9b00313.

[30] M. R. Maser, A. Y. Cui, S. Ryou, T. J. DeLano, Y. Yue, Multilabel classification models for the prediction of cross-coupling reaction conditions, Journal of Chemical Information and Modeling 61 (1) (2021) 156–166. doi:10.1021/acs.jcim.0c01234.

[31] C. W. Coley, R. Barzilay, T. S. Jaakkola, W. H. Green, K. F. Jensen, Prediction of organic reaction outcomes using machine learning, ACS central science 3 (5) (2017) 434–443. doi:10.1021/acscentsci.7b00064.

[32] S. Lack, P. Sobociński, Adhesive categories, in: I. Walukiewicz (Ed.), Foundations of Software Science and Computation Structures, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 273–288. doi:10.1007/978-3-540-24727-2_20.

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

[34] H. Ehrig, U. Golas, F. Hermann, Categorical frameworks for graph transformation and HLR systems based on the DPO approach, Bull. EATCS 102 (2010) 111–121.

[35] A. Habel, D. Plump, $\mathcal{M},\mathcal{N}$-adhesive transformation systems, in: H. Ehrig, G. Engels, H.-J. Kreowski, G. Rozenberg (Eds.), Graph Transformations, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 218–233. doi:10.1007/978-3-642-33654-6_15.

[36] D. Castelnovo, F. Gadducci, M. Miculan, A new criterion for $\mathcal{M},\mathcal{N}$-adhesivity, with an application to hierarchical graphs, in: P. Bouyer, L. Schröder (Eds.), Foundations of Software Science and Computation Structures, Springer International Publishing, Cham, 2022, pp. 205–224. doi:10.1007/978-3-030-99253-8_11.

[37] D. Castelnovo, M. Miculan, On the axioms of $\mathcal{M},\mathcal{N}$-adhesive categories (2024). arXiv:2401.12638.

[38] J. L. Andersen, C. Flamm, D. Merkle, P. F. Stadler, Inferring chemical reaction patterns using rule composition in graph grammars, Journal of Systems Chemistry 4 (1) (2013) 1–4. doi:10.1186/1759-2208-4-4.

[39] P. Selinger, Dagger compact closed categories and completely positive maps: (extended abstract), Electronic Notes in Theoretical Computer Science 170 (2007) 139–163, proceedings of the 3rd International Workshop on Quantum Programming Languages (QPL 2005). doi:10.1016/j.entcs.2006.12.018.

[40] C. Heunen, J. Vicary, Categories for Quantum Theory: An Introduction, Oxford University Press, 2019. doi:10.1093/oso/9780198739623.001.0001.

[41] B. Fong, D. I. Spivak, R. Tuyéras, Backprop as functor: A compositional perspective on supervised learning (2019). doi:10.48550/arXiv.1711.10455.

[42] H. Ehrig, K. Ehrig, A. Habel, K.-H. Pennemann, Constraints and application conditions: From graphs to high-level structures, in: H. Ehrig, G. Engels, F. Parisi-Presicce, G. Rozenberg (Eds.), Graph Transformations, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004, pp. 287–303. doi:10.1007/978-3-540-30203-2_21.

[43] R. Machado, L. Ribeiro, R. Heckel, Rule-based transformation of graph rewriting rules: Towards higher-order graph grammars, Theoretical Computer Science 594 (2015) 1–23. doi:10.1016/j.tcs.2015.01.034.

[44] D. Weininger, SMILES, a chemical language and information system. 1. introduction to methodology and encoding rules, Journal of Chemical Information and Computer Sciences 28 (1) (1988) 31–36. doi:10.1021/ci00057a005.

[45] SMILES tutorial, Website: https://daylight.com/dayhtml_tutorials/languages/smiles/index.html. Accessed 13.07.2024. (1997-2022).

[46] P. Sobocinski, P. Wilson, F. Zanasi, Cartographer: a tool for string diagrammatic reasoning, in: M. Roggenbach, A. Sokolova (Eds.), 8th Conference on Algebra and Coalgebra in Computer Science (CALCO), Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2019, pp. 20:1–20:7.
URL https://cartographer.id/cartographer-calco-2019.pdf

[47] M. Halter, E. Patterson, A. Baas, J. Fairbanks, Compositional scientific computing with catlab and SemanticModels (2020). arXiv:2005.04831.

[48] AlgebraicJulia, Website: https://www.algebraicjulia.org/. Accessed 13.07.2024. (2023).

[49] Kappa language, Website: https://kappalanguage.org/. Accessed 13.07.2024. (2024).

[50] V. Danos, C. Laneve, Formal molecular biology, Theoretical computer science 325 (1) (2004) 69–110. doi:10.1016/j.tcs.2004.03.065.

[51] J. Krivine, Systems biology, ACM SIGLOG News 4 (3) (2017) 43–61. doi:10.1145/3129173.3129182.

[52] N. Behr, J. Krivine, J. L. Andersen, D. Merkle, Rewriting theory for the life sciences: A unifying theory of CTMC semantics, Theor. Comput. Sci. 884 (2021) 68–115. doi:10.1016/j.tcs.2021.07.026.

[53] MØD, Website: https://cheminf.imada.sdu.dk/mod/. Accessed 13.07.2024. (2024).

[54] J. L. Andersen, C. Flamm, D. Merkle, P. F. Stadler, An intermediate level of abstraction for computational systems chemistry, Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 375 (2109) (2017) 20160354. doi:10.1098/rsta.2016.0354.

[55] J. L. Andersen, C. Flamm, D. Merkle, P. F. Stadler, Chemical transformation motifs – modelling pathways as integer hyperflows, IEEE/ACM transactions on computational biology and bioinformatics 16 (2) (2019) 510–523. doi:10.1109/TCBB.2017.2781724.

[56] J. Cockett, S. Lack, Restriction categories i: categories of partial maps, Theoretical Computer Science 270 (1) (2002) 223–259. doi:10.1016/S0304-3975(00)00382-0.

Appendix A. *ICE*-form

Here we give the detailed inductive proof of the fact that any term has an equivalent term in an ICE-form (Proposition 6.8).

Lemma Appendix A.1. *Let* $\mathtt{t}$ *be a term such that the term* $\mathtt{t}; I^{uv}$ *is defined. Then there exists a term* $\mathtt{t}'$ *such that* $\mathtt{t}$ *and* $\mathtt{t}'$ *have the same number of* I*-terms, and one of the following holds:*

(1) $\mathtt{t}; I^{uv} \equiv \mathtt{t}'$*, or*

(2) there is a disconnection I^{ab} *such that* $\mathtt{t}; I^{uv} \equiv I^{ab}; \mathtt{t}'$*.*

Proof. We proceed by induction on the structure of $\mathtt{t}$.

Base cases.

$$
\begin{aligned}
\mathrm{id}; I^{uv} &\equiv I^{uv}; \mathrm{id}, \\
S^w; I^{uv} &\equiv I^{uv}; S^w, && \text{(by (20))} \\
R^{w \mapsto z}; I^{uv} &\equiv I^{uv}; R^{w \mapsto z}, && \text{(by (7))} \\
I^{wz}; I^{uv} &\quad \text{is already in the right form} \\
C^{wz}_{ab}; I^{uv} &\equiv I^{uv}; C^{wz}_{ab}, && \text{(by (24))} \\
E^{w}_{ab}; I^{uv} &\equiv I^{uv}; E^{w}_{ab}, && \text{(by (25))} \\
E^{wz}; I^{uv} &\equiv I^{uv}; E^{wz}, && \text{(by (26))} \\
\bar{I}^{wz}; I^{uv} &\equiv \begin{cases} S^u; S^v \text{ if } w = u, z = v \\ I^{uv}; \bar{I}^{wz} \text{ otherwise,} \end{cases} && \text{(by (15) and (17))} \\
\bar{C}^{wz}_{ab}; I^{uv} &\equiv I^{uv}; \bar{C}^{wz}_{ab}, && \text{(by (29))} \\
\bar{E}^{w}_{ab}; I^{uv} &\equiv I^{uv}; \bar{E}^{w}_{ab}, && \text{(by (28))} \\
\bar{E}^{wz}; I^{uv} &\equiv I^{uv}; \bar{E}^{wz}. && \text{(by (27))}
\end{aligned}
$$

Inductive case. Let $\mathtt{t} : A \to B$ and $\mathtt{s} : B \to C$ be terms such that the statement of the lemma holds. Suppose that the term $\mathtt{t}; \mathtt{s}; I^{uv}$ is defined. Then also the term $\mathtt{s}; I^{uv}$ is defined, so by the inductive hypothesis for $\mathtt{s}$, there is a term $\mathtt{s}'$ with the same number of I-terms as $\mathtt{s}$ such that either (1) $\mathtt{s}; I^{uv} \equiv \mathtt{s}'$, or (2) $\mathtt{s}; I^{uv} \equiv I^{ab}; \mathtt{s}'$ for some I-term I^{ab}. In the first case, we have

$$\mathtt{t}; \mathtt{s}; I^{uv} \equiv \mathtt{t}; \mathtt{s}',$$

and since $\mathtt{t}; \mathtt{s}'$ has the same number of I-terms as $\mathtt{t}; \mathtt{s}$, it is the sought-after term for the inductive case satisfying (1). In the second case, we have

$$\mathtt{t}; \mathtt{s}; I^{uv} \equiv \mathtt{t}; I^{ab}; \mathtt{s}',$$

so that $\mathtt{t}; I^{ab}$ is defined. By the inductive hypothesis for $\mathtt{t}$, there is a term $\mathtt{t}'$ with the same number of I-terms as $\mathtt{t}$ such that either (1) $\mathtt{t}; I^{ab} \equiv \mathtt{t}'$, or (2) $\mathtt{t}; I^{ab} \equiv I^{wz}; \mathtt{t}'$ for some I-term I^{wz}. In the first case, we get

$$\mathtt{t}; \mathtt{s}; I^{uv} \equiv \mathtt{t}'; \mathtt{s}',$$

satisfying (1) for the inductive case, as $\mathtt{t}'; \mathtt{s}'$ and $\mathtt{t}; \mathtt{s}$ have the same number of I-terms. In the second case, we obtain

$$\mathtt{t}; \mathtt{s}; I^{uv} \equiv I^{wz}; \mathtt{t}'; \mathtt{s}',$$

satisfying (2) for the inductive case. □

Corollary Appendix A.2. *Any term is equal to a term of the form* $\mathtt{I};\mathtt{t}$*, where* $\mathtt{I}$ *is a sequence of* I*-terms, and the term* $\mathtt{t}$ *contains no* I*-terms.*

Lemma Appendix A.3. *Let* $\mathtt{t}$ *be a term not containing any* I*-terms such that the term* $\mathtt{t};C^{uv}_{ab}$ *is defined. Then there exists a term* $\mathtt{t}'$ *not containing any* I*-terms such that* $\mathtt{t}$ *and* $\mathtt{t}'$ *have the same number of* C*-terms, and one of the following holds:*

(1) $\mathtt{t};C^{uv}_{ab} \equiv \mathtt{t}'$*, or*

(2) there is a disconnection C^{wz}_{cd} *such that* $\mathtt{t};C^{uv}_{ab} \equiv C^{wz}_{cd};\mathtt{t}'$*.*

Proof. By induction on $\mathtt{t}$.

Base cases.

$$
\begin{aligned}
\mathrm{id};C^{uv}_{ab} &\equiv C^{uv}_{ab};\mathrm{id}, \\
S^w;C^{uv}_{ab} &\equiv C^{uv}_{ab};S^w, && \text{(by (20))} \\
R^{w\mapsto z};C^{uv}_{ab} &\equiv \begin{cases} C^{uv}_{ab};R^{w\mapsto z} \text{ if } w \neq \{a,b\}, \\ C^{uv}_{kb};R^{a\mapsto z};R^{k\mapsto a} \text{ if } w = a, \\ C^{uv}_{ak};R^{b\mapsto z};R^{k\mapsto b} \text{ if } w = b, \end{cases} && \text{(by (7) and (9))} \\
C^{wz}_{cd};C^{uv}_{ab} &\quad \text{is already in the right form}, \\
E^w_{cd};C^{uv}_{ab} &\equiv C^{uv}_{ab};E^w_{cd}, && \text{(by (30))} \\
E^{wz};C^{uv}_{ab} &\equiv C^{uv}_{ab};E^{wz}, && \text{(by (31))} \\
\bar{I}^{wz};C^{uv}_{ab} &\equiv C^{uv}_{ab};\bar{I}^{wz}, && \text{(by (29))} \\
\bar{C}^{wz}_{cd};C^{uv}_{ab} &\equiv \begin{cases} S^w;S^z;R^{c\mapsto j};R^{d\mapsto b};R^{j\mapsto a} \text{ if } w=u, z=v, \\ S^w;S^z;R^{c\mapsto j};R^{d\mapsto a};R^{j\mapsto b} \text{ if } w=v, z=u, \\ C^{uv}_{ij};\bar{C}^{wz}_{cd};R^{i\mapsto a};R^{j\mapsto b} \text{ otherwise}, \end{cases} \\
& && \text{(by (36) and (10))} \\
\bar{E}^w_{cd};C^{uv}_{ab} &\equiv C^{uv}_{ij};\bar{E}^w_{cd};R^{i\mapsto a};R^{j\mapsto b}, && \text{(by (10))} \\
\bar{E}^{wz};C^{uv}_{ab} &\equiv C^{uv}_{ab};\bar{E}^{wz}. && \text{(by (32))}
\end{aligned}
$$

The inductive case is very similar to that of Lemma Appendix A.1. □

Corollary Appendix A.4. *Any term is equal to a term of the form* $\mathtt{I};\mathtt{C};\mathtt{t}$*, where* $\mathtt{I}$ *and* $\mathtt{C}$ *are sequences of* I*-terms and* C*-terms, and the term* $\mathtt{t}$ *contains no* I*-terms or* C*-terms.*

Lemma Appendix A.5. *Let* $\mathtt{t}$ *be a term not containing any* I*- or* C*-terms such that the term* $\mathtt{t};E^u_{ab}$ *is defined. Then there exists a term* $\mathtt{t}'$ *not containing any* I*- or* C*-terms such that* $\mathtt{t}$ *and* $\mathtt{t}'$ *have the same number of* $E^{<0}$*-terms, and one of the following holds:*

(1) $\mathtt{t};E^u_{ab} \equiv \mathtt{t}'$*, or*

(2) there is a disconnection E^w_{cd} *such that* $\mathtt{t};E^u_{ab} \equiv E^w_{cd};\mathtt{t}'$*.*

Proof. By induction on $\mathtt{t}$.

Base cases.

$$\begin{aligned}
\mathrm{id}; E^u_{ab} &\equiv E^u_{ab}; \mathrm{id}, \\
S^w; E^u_{ab} &\equiv E^u_{ab}; S^w, && \text{(by (20))} \\
R^{w\mapsto z}; E^u_{ab} &\equiv \begin{cases} E^u_{ab}; R^{w\mapsto z} \text{ if } w \neq \{a,b\}, \\ E^u_{kb}; R^{a\mapsto z}; R^{k\mapsto a} \text{ if } w = a, \\ E^u_{ak}; R^{b\mapsto z}; R^{k\mapsto b} \text{ if } w = b, \end{cases} && \text{(by (7) and (9))} \\
E^w_{cd}; E^u_{ab} &\quad \text{is already in the right form,} \\
E^{wz}; E^u_{ab} &\equiv E^u_{ab}; E^{wz}, && \text{(by (33))} \\
\bar{I}^{wz}; E^u_{ab} &\equiv E^u_{ab}; \bar{I}^{wz}, && \text{(by (28))} \\
\bar{C}^{wz}_{cd}; E^u_{ab} &\equiv E^u_{ij}; \bar{C}^{wz}_{cd}; R^{i\mapsto a}; R^{j\mapsto b}, && \text{(by (10))} \\
\bar{E}^w_{cd}; E^u_{ab} &\equiv \begin{cases} S^w; R^{c\mapsto j}; R^{d\mapsto b}; R^{j\mapsto a} \text{ if } w = u, \\ E^u_{ij}; \bar{E}^u_{cd}; R^{i\mapsto a}; R^{j\mapsto b} \text{ otherwise,} \end{cases} && \text{(by (36) and (10))} \\
\bar{E}^{wz}; E^u_{ab} &\equiv E^u_{ab}; \bar{E}^{wz}. && \text{(by (34))}
\end{aligned}$$

The inductive case is very similar to that of Lemma Appendix A.1. □

Corollary Appendix A.6. *Any term is equal to a term of the form* $\mathtt{I}; \mathtt{C}; \mathtt{E}^{<0}; \mathtt{t}$*, where* $\mathtt{I}$*,* $\mathtt{C}$ *and* $\mathtt{E}^{<0}$ *are sequences of* I*-,* C*-, and* $E^{<0}$*-terms, and the term* $\mathtt{t}$ *contains no* I*-,* C*-, or* $E^{<0}$*-terms.*

Lemma Appendix A.7. *Let* $\mathtt{t}$ *be a term not containing any* I*-,* C*-, or* $E^{<0}$*-terms such that the term* $\mathtt{t}; E^{uv}$ *is defined. Then there exists a term* $\mathtt{t}'$ *not containing any* I*-,* C*-, or* $E^{<0}$*-terms such that* $\mathtt{t}$ *and* $\mathtt{t}'$ *have the same number of* $E^{\geq 0}$*-terms, and one of the following holds:*

(1) $\mathtt{t}; E^{uv} \equiv \mathtt{t}'$*, or*

(2) there is a disconnection E^{wz} *such that* $\mathtt{t}; E^{uv} \equiv E^{wz}; \mathtt{t}'$*.*

Proof. By induction on $\mathtt{t}$.

Base cases.

$$\begin{aligned}
\mathrm{id}; E^{uv} &\equiv E^{uv}; \mathrm{id}, \\
S^w; E^{uv} &\equiv E^{uv}; S^w, && \text{(by (20))} \\
R^{w\mapsto z}; E^{uv} &\equiv \begin{cases} E^{uv}; R^{w\mapsto z} \text{ if } z \neq v, \\ E^{uw}; R^{w\mapsto v} \text{ if } z = v, \end{cases} && \text{(by (7) and (8))} \\
E^{wz}; E^{uv} &\quad \text{is already in the right form,} \\
\bar{I}^{wz}; E^{uv} &\equiv E^{uv}; \bar{I}^{wz}, && \text{(by (27))} \\
\bar{C}^{wz}_{cd}; E^{uv} &\equiv E^{uv}; \bar{C}^{wz}_{cd}, && \text{(by (32))} \\
\bar{E}^w_{cd}; E^{uv} &\equiv E^{uv}; \bar{E}^w_{cd}, && \text{(by (34))} \\
\bar{E}^{wz}; E^{uv} &\equiv \begin{cases} S^u; S^v \text{ if } w = u \text{ and } z = v, \\ E^{uv}; \bar{E}^{wz} \text{ otherwise.} \end{cases} && \text{(by (15) and (17))}
\end{aligned}$$

The inductive case is very similar to that of Lemma Appendix A.1. □

Corollary Appendix A.8. *Any term is equal to a term of the form*

$$\mathtt{I};\mathtt{C};\mathtt{E}^{<0};\mathtt{E}^{\geq 0};\mathtt{t},$$

where $\mathtt{I}$, $\mathtt{C}$, $\mathtt{E}^{<0}$ *and* $\mathtt{E}^{\geq 0}$ *are sequences of* I*-,* C*-,* $E^{<0}$*, and* $E^{\geq 0}$*-terms, and the term* $\mathtt{t}$ *contains no* I*-,* C*-,* $E^{<0}$*, or* $E^{\geq 0}$*-terms.*

Lemma Appendix A.9. *Let* $\mathtt{t}$ *be a term not containing any* I*-,* C*- or* E*-terms such that the term* $\mathtt{t};\bar{E}^{uv}$ *is defined. Then there exists a term* $\mathtt{t}'$ *not containing any* I*-,* C*- or* E*-terms such that* $\mathtt{t}$ *and* $\mathtt{t}'$ *have the same number of* $\bar{E}^{\geq}$*-terms, and one of the following holds:*

(1) $\mathtt{t};\bar{E}^{uv} \equiv \mathtt{t}'$*, or*

(2) there is a connection $\bar{E}^{wz}$ *such that* $\mathtt{t};\bar{E}^{uv} \equiv \bar{E}^{wz};\mathtt{t}'$*.*

Proof. By induction on $\mathtt{t}$.

Base cases.

$$\begin{aligned}
\mathrm{id};\bar{E}^{uv} &\equiv \bar{E}^{uv};\mathrm{id}, && \\
S^w;\bar{E}^{uv} &\equiv \bar{E}^{uv};S^w, && \text{(by (20))}\\
R^{w\mapsto z};\bar{E}^{uv} &\equiv \begin{cases} \bar{E}^{uv};R^{w\mapsto z} \text{ if } z \neq v, \\ \bar{E}^{uw};R^{w\mapsto v} \text{ if } z = v, \end{cases} && \text{(by (7) and (8))}\\
\bar{I}^{wz};\bar{E}^{uv} &\equiv \bar{E}^{uv};\bar{I}^{wz}, && \text{(by (26))}\\
\bar{C}^{wz}_{cd};\bar{E}^{uv} &\equiv \bar{E}^{uv};\bar{C}^{wz}_{cd}, && \text{(by (31))}\\
\bar{E}^{w}_{cd};\bar{E}^{uv} &\equiv \bar{E}^{uv};\bar{E}^{w}_{cd}, && \text{(by (33))}\\
\bar{E}^{wz};\bar{E}^{uv} & \quad \text{is already in the right form.} &&
\end{aligned}$$

The inductive case is very similar to that of Lemma Appendix A.1. □

Corollary Appendix A.10. *Any term is equal to a term of the form*

$$\mathtt{I};\mathtt{C};\mathtt{E}^{<0};\mathtt{E}^{\geq 0};\bar{\mathtt{E}}^{\geq 0};\mathtt{t},$$

where the term $\mathtt{t}$ *contains no* I*-,* C*-,* E*-, or* $\bar{E}^{\geq 0}$*-terms.*

Lemma Appendix A.11. *Let* $\mathtt{t}$ *be a term not containing any* I*-,* C*-,* E*-, or* $\bar{E}^{\geq 0}$*-terms such that the term* $\mathtt{t};\bar{E}^{u}_{ab}$ *is defined. Then there exists a term* $\mathtt{t}'$ *not containing any* I*-,* C*-,* E*-, or* $\bar{E}^{\geq 0}$*-terms such that* $\mathtt{t}$ *and* $\mathtt{t}'$ *have the same number of* $\bar{E}^{<0}$*-terms, and one of the following holds:*

(1) $\mathtt{t};\bar{E}^{u}_{ab} \equiv \mathtt{t}'$*, or*

(2) there is a connection $\bar{E}^{w}_{cd}$ *such that* $\mathtt{t};\bar{E}^{u}_{ab} \equiv \bar{E}^{w}_{cd};\mathtt{t}'$*.*

Proof. By induction on $\mathtt{t}$.

Base cases.

$$\begin{aligned}
\mathrm{id};\bar{E}^u_{ab} &\equiv \bar{E}^u_{ab};\mathrm{id},\\
S^w;\bar{E}^u_{ab} &\equiv \bar{E}^u_{ab};S^w, && \text{(by (20))}\\
R^{w\mapsto z};\bar{E}^u_{ab} &\equiv \begin{cases} \bar{E}^u_{ab};R^{w\mapsto z} \text{ if } z\notin\{a,b\},\\ \bar{E}^u_{wb} \text{ if } z=a,\\ \bar{E}^u_{aw} \text{ if } z=b, \end{cases} && \text{(by (7) and (9))}\\
\bar{I}^{wz};\bar{E}^u_{ab} &\equiv \bar{E}^u_{ab};\bar{I}^{wz}, && \text{(by (25))}\\
\bar{C}^{wz}_{cd};\bar{E}^u_{ab} &\equiv \bar{E}^u_{ab};\bar{C}^{wz}_{cd}, && \text{(by (30))}\\
\bar{E}^w_{cd};\bar{E}^u_{ab} &\quad \text{is already in the right form.}
\end{aligned}$$

The inductive case is very similar to that of Lemma Appendix A.1. □

Corollary Appendix A.12. *Any term is equal to a term of the form*

$$\mathtt{I};\mathtt{C};\mathtt{E}^{<0};\mathtt{E}^{\geq 0};\bar{\mathtt{E}}^{\geq 0};\bar{\mathtt{E}}^{<0};\mathtt{t},$$

where the term $\mathtt{t}$ *contains no I-, C-, E- or* $\bar{E}$*-terms.*

Lemma Appendix A.13. *Let* $\mathtt{t}$ *be a term not containing any I-, C-, E- or* $\bar{E}$*-terms such that the term* $\mathtt{t};\bar{C}^{uv}_{ab}$ *is defined. Then there exists a term* $\mathtt{t}'$ *not containing any I-, C-, E- or* $\bar{E}$*-terms such that* $\mathtt{t}$ *and* $\mathtt{t}'$ *have the same number of* $\bar{C}$*-terms, and one of the following holds:*

(1) $\mathtt{t};\bar{C}^{uv}_{ab}\equiv\mathtt{t}'$*, or*

(2) there is a connection $\bar{C}^{wz}_{cd}$ *such that* $\mathtt{t};\bar{C}^{uv}_{ab}\equiv\bar{C}^{wz}_{cd};\mathtt{t}'$.

Proof. By induction on $\mathtt{t}$.

Base cases.

$$\begin{aligned}
\mathrm{id};\bar{C}^{uv}_{ab} &\equiv \bar{C}^{uv}_{ab};\mathrm{id},\\
S^w;\bar{C}^{uv}_{ab} &\equiv \bar{C}^{uv}_{ab};S^w, && \text{(by (20))}\\
R^{w\mapsto z};\bar{C}^{uv}_{ab} &\equiv \begin{cases} \bar{C}^{uv}_{ab};R^{w\mapsto z} \text{ if } z\notin\{a,b\},\\ \bar{C}^{uv}_{wb} \text{ if } z=a,\\ \bar{C}^{uv}_{aw} \text{ if } z=b, \end{cases} && \text{(by (7) and (9))}\\
\bar{I}^{wz};\bar{C}^{uv}_{ab} &\equiv \bar{C}^{uv}_{ab};\bar{I}^{wz}, && \text{(by (24))}\\
\bar{C}^{wz}_{cd};\bar{C}^{uv}_{ab} &\quad \text{is already in the right form.}
\end{aligned}$$

The inductive case is very similar to that of Lemma Appendix A.1. □

Corollary Appendix A.14. *Any term is equal to a term of the form*

$$\mathtt{I};\mathtt{C};\mathtt{E}^{<0};\mathtt{E}^{\geq 0};\bar{\mathtt{E}}^{\geq 0};\bar{\mathtt{E}}^{<0};\bar{\mathtt{C}};\mathtt{t},$$

where the term $\mathtt{t}$ *contains no I-, C-, E-,* $\bar{E}$*- or* $\bar{C}$*-terms.*

Lemma Appendix A.15. *Let* $\mathtt{t}$ *be a term not containing any I-, C-, E-,* $\bar{E}$*- or* $\bar{C}$*-terms such that the term* $\mathtt{t};\bar{I}^{uv}$ *is defined. Then there exists a term* $\mathtt{t}'$ *not containing any I-, C-, E-,* $\bar{E}$*- or* $\bar{C}$*-terms such that* $\mathtt{t}$ *and* $\mathtt{t}'$ *have the same number of* $\bar{I}$*-terms, and one of the following holds:*

(1) $\mathtt{t}; \bar{I}^{uv} \equiv \mathtt{t}'$*, or*

(2) there is a connection $\bar{I}^{ab}$ *such that* $\mathtt{t}; \bar{I}^{uv} \equiv \bar{I}^{ab}; \mathtt{t}'$.

Proof. By induction on $\mathtt{t}$.

Base cases.

$$\begin{aligned}
\mathrm{id}; \bar{I}^{uv} &\equiv \bar{I}^{uv}; \mathrm{id}, \\
S^w; \bar{I}^{uv} &\equiv \bar{I}^{uv}; S^w, && \text{(by (20))} \\
R^{w \mapsto z}; \bar{I}^{uv} &\equiv \bar{I}^{uv}; R^{w \mapsto z}, && \text{(by (7))} \\
\bar{I}^{wz}; \bar{I}^{uv} &\quad \text{is already in the right form.}
\end{aligned}$$

The inductive case is very similar to that of Lemma Appendix A.1. □

Corollary Appendix A.16. *Any term is equal to a term of the form*

$$\mathtt{I}; \mathtt{C}; \mathtt{E}^{<0}; \mathtt{E}^{\geq 0}; \bar{\mathtt{E}}^{\geq 0}; \bar{\mathtt{E}}^{<0}; \bar{\mathtt{C}}; \bar{\mathtt{I}}; \mathtt{t},$$

where the term $\mathtt{t}$ *contains only S-, R-, and identity terms.*

Lemma Appendix A.17. *Let* $\mathtt{t}$ *be a term containing only S-, R-, and identity terms such that the term* $S^u; \mathtt{t}$ *is defined. Then there exists a term* $\mathtt{t}'$ *containing only S-, R-, and identity terms such that* $\mathtt{t}$ *and* $\mathtt{t}'$ *have the same number of S-terms, and one of the following holds:*

(1) $S^u; \mathtt{t} \equiv \mathtt{t}'$*, or*

(2) there is a term S^v *such that* $S^u; \mathtt{t} \equiv \mathtt{t}'; S^v$.

Proof. By induction on $\mathtt{t}$.

Base cases.

$$\begin{aligned}
S^u; \mathrm{id} &\equiv \mathrm{id}; S^u, \\
S^u; S^w &\quad \text{is already in the right form,} \\
S^u; R^{w \mapsto z} &\equiv \begin{cases} R^{w \mapsto z}; S^u \text{ if } u \neq w, \\ R^{u \mapsto z} \text{ if } u = w. \end{cases} && \text{(by (5) and (6))}
\end{aligned}$$

The inductive case is very similar to that of Lemma Appendix A.1. □

Corollary Appendix A.18 (Proposition 6.8)**.** *Any term is equal to a term in an ICE-form.*