

# Chemical Graph Transformation with Stereo-Information

Jakob Lykke Andersen<sup>1,9</sup>(✉), Christoph Flamm<sup>2,8</sup>, Daniel Merkle<sup>1</sup>(✉),  
and Peter F. Stadler<sup>2,3,4,5,6,7</sup>

<sup>1</sup> Department of Mathematics and Computer Science,  
University of Southern Denmark, 5230 Odense, Denmark  
{jlandersen,daniel}@imada.sdu.dk

<sup>2</sup> Institute for Theoretical Chemistry, University of Vienna, 1090 Wien, Austria  
xtof@tbi.univie.ac.at

<sup>3</sup> Bioinformatics Group, Department of Computer Science,  
and Interdisciplinary Center for Bioinformatics,  
University of Leipzig, 04107 Leipzig, Germany  
stadler@bioinf.uni-leipzig.de

<sup>4</sup> Max Planck Institute for Mathematics in the Sciences, 04103 Leipzig, Germany

<sup>5</sup> Fraunhofer Institute for Cell Therapy and Immunology, 04103 Leipzig, Germany

<sup>6</sup> Center for Non-coding RNA in Technology and Health, University of Copenhagen,  
1870 Frederiksberg, Denmark

<sup>7</sup> Santa Fe Institute, 1399 Hyde Park Rd, Santa Fe, NM 87501, USA

<sup>8</sup> Research Network Chemistry Meets Microbiology,  
University of Vienna, 1090 Wien, Austria

<sup>9</sup> Tokyo Institute of Technology, Earth-Life Science Institute, Tokyo 152-8550, Japan  
jlandersen@elsi.jp

**Abstract.** Double Pushout graph transformation naturally facilitates the modelling of chemical reactions: labelled undirected graphs model molecules and direct derivations model chemical reactions. However, the most straightforward modelling approach ignores the relative placement of atoms and their neighbours in space. Stereoisomers of chemical compounds thus cannot be distinguished, even though their chemical activity may differ substantially. In this contribution we propose an extended chemical graph transformation system with attributes that encode information about local geometry. The modelling approach is based on the so-called “ordered list method”, where an order is imposed on the set of incident edges of each vertex, and permutation groups determine equivalence classes of orderings that correspond to the same local spatial embedding. This method has previously been used in the context of graph transformation, but we here propose a framework that also allows for partially specified stereoinformation. While there are several stereochemical configurations to be considered, we focus here on the tetrahedral molecular shape, and suggest general principles for how to treat all other chemically relevant local geometries. We illustrate our framework using several chemical examples, including the enumeration of stereoisomers of carbohydrates and the stereospecific reaction for the aconitase enzyme in the citric acid cycle.

**Keywords:** Double Pushout · Chemical graph transformation system · Stereochemistry

## 1 Introduction

Graph transformation systems have a long history in molecular biology [24]. Applications to chemical reaction systems have evolved from abstract artificial chemistry models such as Fontana’s AlChemY [13, 14] based on lambda calculus. An early attempt at more realistic modelling of chemistry with graph transformation [6] and an early perspectives article [29] proposed a variety of potential applications.

Although general graph transformation tools, such as AGG [26], have also been used to implement models of chemical systems [10], there is one crucial aspect where chemistry differs from the usual setup in the graph transformation literature. The latter focusses on rewriting a single (usually connected) graph, thus yielding a traditional formal language. Chemical reactions, in contrast, usually involve multiple molecules; chemical graph transformations therefore operate on *multisets* of graphs to produce a chemical “space” or “universe” [4], see also [17] for a similar construction in the context of DNA computing. With the software package MØD [2] we have developed a versatile suite for working with this type of transformation [5]. The package handles composition of rules and provides a domain specific language for graph language generation [3, 4].

Mathematical models for molecular compounds may be specified at different levels of abstraction. At the coarsest, arithmetical level molecular formulas describe only the number and type of constituent atoms; a finer topological level uses graphs to determine the adjacencies between atoms; a further refinement also determines the (relative) spatial arrangement of atoms and thus the molecule’s geometry. Stereoisomers, that is, molecules with the same topology but different geometry, often have similar physical and chemical properties but differ dramatically in their biological and pharmacological activity. A famous example is the sedative thalidomide. The compound with the German trade name Contergan has a sedative effect. Its non-superposable mirror image (such a pair of compounds is called enantiomorphic), however, causes severe birth defects. The stereospecific — and in particular enantioselective — synthesis of such compounds is a very challenging task in practice. In order for a graph transformation model of chemistry to be useful in practical applications, it therefore needs to be able to properly model stereoisomers and stereospecific chemical reactions. This task is not made simpler by the fact that stereochemical terms are often not well-defined in a mathematical sense [16].

To date, most chemical graph transformation models, with the notable exception is the hypergraph rewriting approach explored in [10], lack support for stereochemistry. The chemical literature, however, has recognized early-on that the ability to handle stereochemistry is a prerequisite for the practical applicability of computational models of chemistry: Already in the sixties of the last century the “ordered list method” was introduced [20, 28]. It exploits the representation

of graphs as adjacency lists by using the ordering of the edge lists to encode geometric information. Alternative approaches rely on transformation of structures to larger, ordinary graphs that encode the stereochemical situations, e.g. [1] or aim at the encoding in the form of linear descriptions such as SMILES [27] or CAST [25]. The chemical literature usually annotates local geometric information in terms of IUPAC nomenclature rules. For example, the local geometry at a tetrahedral centre is determined as “R” or “S” depending on a complex set of inherently non-local precedence rules for the four neighbours [8]. Such representation of geometric information is not designed to allow the implementation of chemical reactions as local rewriting operations.

Here we advocate a strategy that differs in a conceptually important point from [10]: Their hypergraph approach explicitly uses transformation rules to generate equivalent tetrahedral centres, which results in exponentially many graphs (in the number of centres) representing the same molecule. Instead, we propose here to incorporate the symmetries that define equivalent local geometries directly into the morphisms themselves. This also allows us to preserve the modelling principle that each graph is equivalent to just one molecule, and that each direct direction is a proper chemical reaction. It is not in all reactions that the (full) geometric information is relevant, and we therefore also introduce a hierarchy of local atom configurations that allows the representation of partially known stereo-information, both in graphs and rules. This approach can be seen as a special case of graph transformation with node inheritance [18], though we opt for a more direct modelling approach, closer to a practical implementation, where the inheritance is captured in an specialised algebra using principles from term algebras.

We introduce stereochemistry and molecular shapes in Sect. 2, and in Sect. 3 we describe the graph model and transformation system with attributes that encodes information about local geometry. We give several application examples in Sect. 4 and conclude with Sect. 5. In the Appendix we present the code used for the application examples.

## 2 Molecular Shapes



The connectivity of molecules can be modelled trivially by undirected graphs, but this ignores the relative placement of atoms and their neighbours in 3D space. An intermediary view is to look locally at each atom and characterise the shape that the incident bonds form. Each atom features (depending on its type) a certain number of valence electrons. Part of these are shared with adjacent atoms in the formation of chemical bonds, while others remain localized at their atom and form so-called lone electron pairs. The Lewis diagram [19] of a molecule describes the distribution of valence electrons into bonding electron pairs and lone pairs. Backed by a grounding in quantum theory, the Gillespie-Nyholm theory, also called the Valence-Shell Electron-Pair Repulsion (VSEPR) theory [15], then explains the local geometry in terms of Lewis formula by means of three simple rules: (1) electron pairs repulse each other and thus attain a geometry

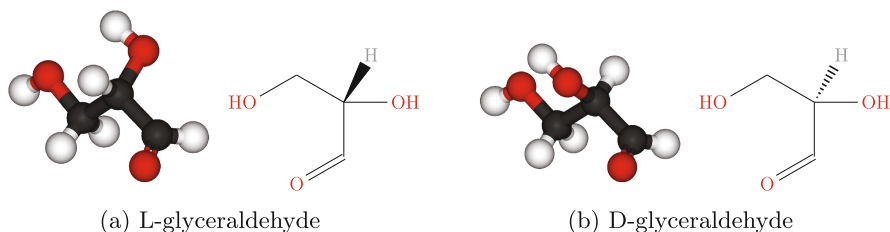
that maximizes their mutual angular distances; (2) double and triple bonds can be treated like single bonds; and (3) lone electron pairs are treated like chemical bonds. Changes in bond orders and/or the number of lone pairs therefore affect the geometry as part of a chemical reaction. The distinction between bonds and lone pair allows the model to define fine-grained shapes, for example:

- The oxygen in a water molecule has 2 lone pairs and 2 incident bonds, giving it the “bent” shape.
- The nitrogen in an ammonia molecule has 1 lone pair and 3 incident bonds, giving it the “trigonal pyramidal” shape.
- The carbon in a methane molecule has no lone pairs and 4 incident bonds, giving the “tetrahedral” shape.

In terms of the VSEPR theory, each of these three examples correspond to a central atom with four neighbours, and the difference in shape arise from distinguishing bonds from lone pairs. Two atoms with the same sum of incident bonds and lone pairs have the same intrinsic geometry, in this case as a tetrahedron with the atom in the centre and the neighbours placed in the corners. In the model we thus only consider the basic shapes, from which the “visible” geometry of the molecule can be recovered by considering the lone pairs.

A comprehensive model of stereochemistry should include separate treatment of each possible shape. In this contribution we focus on the tetrahedral shape and the general modelling framework that also allows for partial specification of stereo-information in transformation rules. Future extensions will then implement the remaining chemically relevant shapes.

Throughout the paper we use the depiction of tetrahedral shapes usually used in chemistry, where wedge (  ) and hash (  ) bonds are used to indicate their 3D embedding. In Fig. 1 this is illustrated on the two stereoisomers of glyceraldehyde.

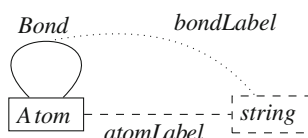


**Fig. 1.** Depiction of the two stereoisomers of glyceraldehyde in 3D (3D depictions from [https://en.wikibooks.org/wiki/Organic\\_Chemistry/Chirality](https://en.wikibooks.org/wiki/Organic_Chemistry/Chirality)) and in 2D with wedge/hash bond notation to indicate the 3D embedding. The broad end of a wedge (resp. hash) bond is placed above (resp. below) the plane of drawing of the narrow end.

### 3 Model

#### 3.1 Molecules as Typed Attributed Graphs

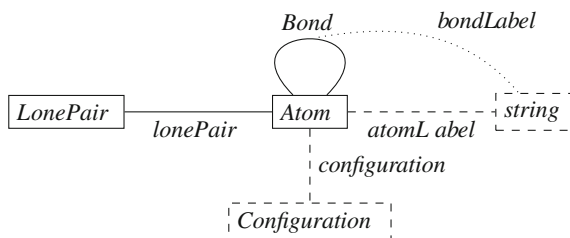
Molecules without stereochemical information can be modelled directly using simple undirected graphs, with labels on vertices and edges. For extending this model we recast the model described in [5] in terms of typed attributed graphs (e.g., see [9]), which simply results in the type graph shown in Fig. 2. In the practical use of a chemical graph transformation system it is useful to enable/disable stereochemical information in different contexts. The stereochemical model therefore only adds to the type graph of the basic model.



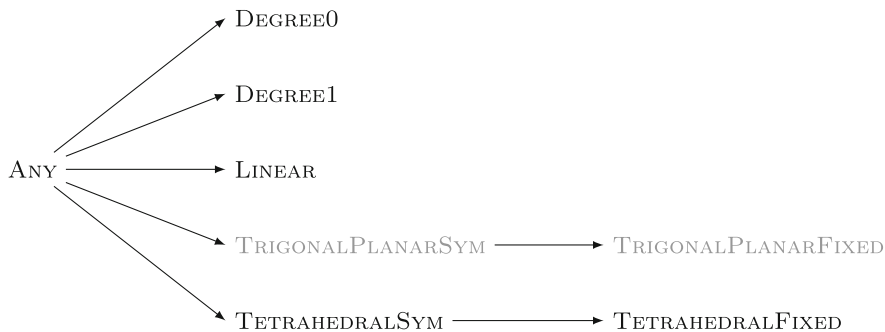
**Fig. 2.** Type graph for the basic molecule model, where each atom vertex and bond edge are attributed with strings, that encode the atom type, charge, and bond order.

Not all combinations of atom types, charges, number of lone pairs, and shapes are chemically valid. However, for simplicity we here present a general model for describing local geometry, and leave out the details of checking for chemical validity. The number of combinations is quite limited and in the end the check can therefore be handled by a moderately sized lookup table.

For representing lone pairs we allow each atom to have additional neighbours of type *LonePair* (see Fig. 3). In the following when we refer to the degree of an atom and its neighbours we thus include the lone pairs. On a practical note, we can simply represent the number of lone pairs at each atom, and adapt the morphism algorithms accordingly.



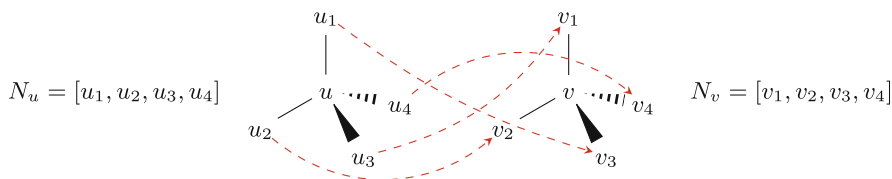
**Fig. 3.** The extended type graph for representing stereochemistry. A new type of vertex is introduced for the modelling of lone electron pairs, and a new atom attribute is added for representing molecular shapes and embeddings into the shapes. Each atom is only allowed to have 1 configuration, while it may have multiple neighbouring lone pairs.



**Fig. 4.** The category of shapes,  $\mathcal{C}_{\text{Shape}}$ , used as a basis for encoding stereochemical configurations. Leaf objects correspond to actual molecular shapes while the remaining objects provide a means for specifying partial stereo-information by acting as “variable” shapes. In particular, the ANY shape is the initial object that acts as an unconstrained variable. The two trigonal planar shapes are shown only as an example of how the category will be extended in the future. They are briefly discussed in the concluding remarks.

Next we introduce a category of shapes  $\mathcal{C}_{\text{Shape}}$ , where the objects and morphisms are explicitly defined, see Fig. 4. In principle we add an object for each general shape described in the VSEPR theory, though here we focus on the tetrahedral shape. We additionally introduce several “variable” shapes for more expressive modelling of transformation rules, including the ANY shape which is the initial object of the category. This allows for the direct expression of (partially) unknown configurations, both in rules and in molecules.

In contrast to the “ordered list method” in [20,28] we do not modify the underlying storage of the graph. Instead we store the neighbour ordering in a *Configuration* attribute on each atom along with the geometric shape of the atom. That is, a configuration is a pair  $\langle S, N \rangle$  of a shape object  $S$  and an ordered list of all neighbours of the atom  $N$ . Most shapes may only be assigned to atoms of a specific degree (see below), e.g., the tetrahedral shape requires the atom to have degree 4. As each configuration references the neighbours in the graph, the definition of configuration morphisms requires an already valid graph morphism, which we assume also to be injective due to the modelling of chemistry [5]. Let  $m: G_1 \rightarrow G_2$  be such an injective typed graph morphism, with respect to all attributes except for the configurations. For deciding whether  $m$  is also valid when taking configurations into account, consider an atom vertex  $u$  of  $G_1$  with configuration  $\langle S_1, N_1 \rangle$ , and its image  $v = m(u)$  with configuration  $\langle S_2, N_2 \rangle$ . We first require that a shape morphism  $S_1 \rightarrow S_2$  exists. Then, from the neighbour lists  $N_1 = [u_1, u_2, \dots, u_{d_1}]$  and  $N_2 = [v_1, v_2, \dots, v_{d_2}]$  create an index map  $m_I: \{1, 2, \dots, d_1\} \rightarrow \{1, 2, \dots, d_2\}$  such that if  $m(u_i) = v_j$  then  $m_I(i) = j$ . Each shape morphism  $S_1 \rightarrow S_2$  may now define additional constraints the index map  $m_I$  must fulfil (see Fig. 5 for an example). Though, for the current set of shapes only morphisms among configurations with TETRAHEDRALFIXED shape has additional constraints.



Induced index map:  $m_I = \{1 \mapsto 3, 2 \mapsto 2, 3 \mapsto 1, 4 \mapsto 4\} = (1\ 3)(2)(4)$

**Fig. 5.** Example of a graph morphism, which is not a valid stereo morphism. The two vertices  $u, v$  both have the TETRAHEDRALFIXED shape, and the indicated neighbour lists  $N_u$  and  $N_v$ . A graph morphism  $m$  is given, indicated by the dashed, red arrows and with  $m(u) = v$ . This induces the index map  $m_I = \{1 \mapsto 3, 2 \mapsto 2, 3 \mapsto 1, 4 \mapsto 4\}$ , i.e., the permutation  $(1\ 3)(2)(4)$ . As this permutation does not describe a symmetry of a tetrahedron, following our encoding convention, the graph morphism is not a valid stereo morphism. (Color figure online)

In the following we describe intended semantics, degree constraints, and index map constraints of each shape.

*The TETRAHEDRALFIXED Shape* can only be attached to atoms of degree 4. We interpret a neighbour list  $[v_1, v_2, v_3, v_4]$  geometrically in the following manner: the neighbours are placed in the corners of a regular tetrahedron, and  $v$  is placed in the centre. When looking from  $v_1$  towards  $v$ , the neighbours  $v_2, v_3, v_4$  appear in counter-clockwise order. With this encoding the symmetries of a tetrahedron can be expressed as the permutation group generated by  $\langle (1\ 2\ 3\ 4), (1\ 2)(3\ 4) \rangle$  acting on the neighbour list, corresponding to the alternating group on 4 elements as expected. A morphism from one TETRAHEDRALFIXED configuration to another thus requires the index map to be a permutation from this group. In Fig. 5 an example of a graph morphism that does not meet this requirement is shown.

*The TETRAHEDRALSYM Shape.* In some cases the specific embedding of an atom in tetrahedral shape is unknown, and in some cases it is beneficial to be able to match both possible tetrahedral embeddings. We therefore introduce this shape that also requires atom degree 4, has the geometric shape of a tetrahedron, but with no particular assignment of neighbours to the corners. The symmetries of the neighbours are therefore the complete symmetric group on 4 elements. As it has a morphism to the TETRAHEDRALFIXED shape it can be used as a restricted “variable” in transformation rules.

*The ANY Shape* has no degree constraints, and all neighbour lists are equivalent. It is the initial object of the shape category, and can therefore be used as an unrestricted “variable” in transformation rules.

*The DEGREE0, DEGREE1, and LINEAR Shapes* require degree 0, 1, and 2, resp., of the atoms they are attached to. Geometrically, an atom with the LINEAR shape is located on the line between its two neighbours.

### 3.2 Transformation Rules and Derivations

For a DPO transformation rule  $p = (L \xleftarrow{l} K \xrightarrow{r} R)$  we already require  $l$  and  $r$  to be graph monomorphisms. In the extension to stereochemical information, we require them to be isomorphisms on the configuration attributes. That is, either an atom has no configuration attribute in  $K$ , or it has the same attribute in  $L$ ,  $K$ , and  $R$ . The top span of Fig. 6 shows an example rule where the change of configuration is combined with partial stereo-information. As configurations contain lists of neighbours in the graph, the isomorphism requirement for configurations implies that only atoms of  $K$  where all incident edges also are in  $K$  can have a configuration attribute. From the perspective of modelling chemistry this means that when bonds are broken or formed, one must be explicit about the change of molecular shape for the incident atoms.

In rule application the configurations with non-leaf shapes (see Fig. 4) act as unnamed variables, similar to transformation with term attributes described in [9]. That is, in the transformation of a graph  $G$  with a rule  $p = (L \xleftarrow{l} K \xrightarrow{r} R)$ , the match morphism  $m: L \rightarrow G$  implicitly determines an assignment of configurations such that substitution yields isomorphic configurations. This is illustrated with both vertex 0 and 1 in the direct derivation shown in Fig. 6. Vertex 1 has an ANY configuration in  $L$ , and is being assigned to a TETRAHEDRALS<sub>Y</sub>M configuration through  $m$ . As it also has this configuration in  $K$  and  $R$ , the pushout requirements preserve the TETRAHEDRALS<sub>Y</sub>M configuration through  $D$  to  $H$ . Vertex 0 has a TETRAHEDRALS<sub>Y</sub>M configuration in  $L$ , which is being assigned to a TETRAHEDRALFIXED configuration. However, the vertex has no configuration in  $K$ , and a new TETRAHEDRALS<sub>Y</sub>M configuration is added in  $R$ . The rule therefore effectively matches any tetrahedron to vertex 0 and generalizes it to a TETRAHEDRALS<sub>Y</sub>M.

## 4 Application Examples

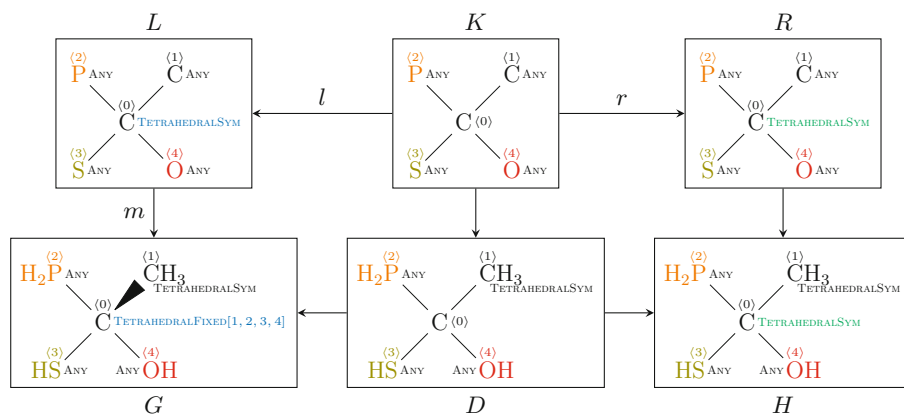
We have extended the graph transformation system of MØD [2, 5] with the model for stereochemistry. Morphisms are found using the VF2 algorithm [7], where shape morphisms are checked during matching. Index map constraints require the complete neighbourhood of a vertex to be mapped to the host graph. For simplicity this check is deferred to after a total morphism has been found.

In the following we illustrate the use of the modelling framework. The code for each example can be found in the appendix, and can be experimented with in the live version of MØD at <http://mod.imada.sdu.dk/playground.html>.

### 4.1 Stereospecific Aconitase

One of the central metabolic pathways is the citric acid cycle, which contains a reaction that converts the molecule citrate into isocitrate. This reaction, facilitated by the aconitase enzyme, is stereospecific which means that it only produces D-isocitrate and not the stereoisomer L-isocitrate. While the modelling of





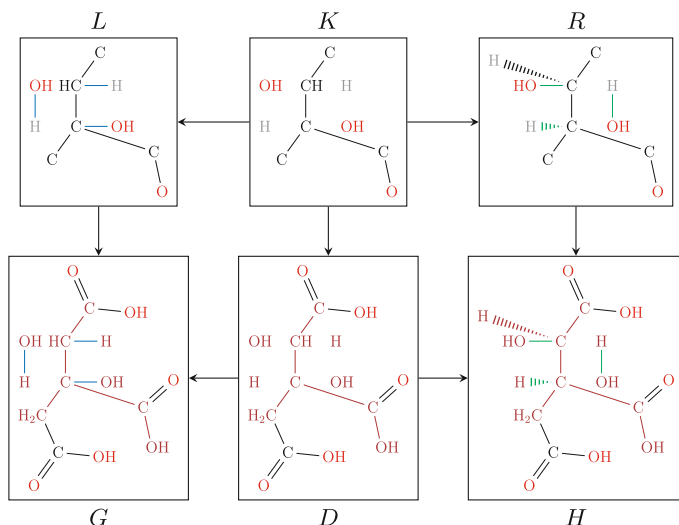
**Fig. 6.** A direct derivation with explicitly annotated configuration data. Vertex 0 and 1 have variable configurations with TETRAHEDRALSYM and ANY shape, such that they can match more specialised configurations. As vertex 1 also has a configuration in *K*, its assigned TETRAHEDRALSYM configuration in *G* is transferred to *D* and *H* as well. The configuration on vertex 0 is on the other hand being deleted and replaced with a new configuration in *R*. The original TETRAHEDRALFIXED configuration in *G* is therefore replaced accordingly.

this reaction as a transformation rule can be done in the hypergraph approach described in [10], the present approach also allows us to generalize the rule to be applicable to molecules other than isocitrate, that share the same context. This is shown in Fig. 7 where a generalized rule for aconitase is shown being applied to citrate and water.

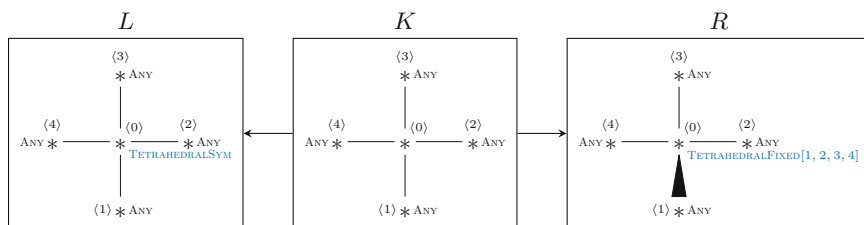
## 4.2 Generation of Stereoisomers

Tartaric acid is the most important chemical compound for the discovery of the concept of chirality. Tartaric acid has three stereoisomers, two are chiral (i.e., their mirror image is non-superposable) and one is achiral (i.e., it equals its mirror image). The crystal structure of the double salt of the stereoisomers of tartaric acid (potassium sodium tartrate tetrahydrate) was analysed by Louis Pasteur. He performed a morphological analysis and analysed the shapes of the different macroscopic crystals. The macroscopic (non-)superposability of the idealised shape of the crystals established the existence of molecular chirality [12].

We use the tartaric acid molecule here as an example to illustrate how all stereoisomers with partial and fully specified stereoinformation can be inferred in the rule-based framework. This is accomplished by repeated application of the rule shown in Fig. 8. As the central atom has TETRAHEDRALSYM shape it can be used to either fixate the tetrahedral embedding or change an existing one. We here also extend the ordinary atom labels to include the special unnamed variable label ‘\*’ that can be assigned any other atom label during matching. Figure 9 shows the result of repeatedly applying the rule to a model of tartaric



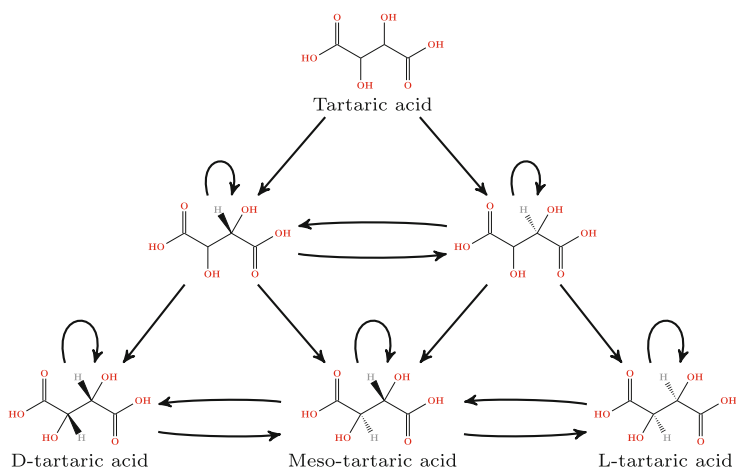
**Fig. 7.** Illustration of a generalized transformation rule for the aconitase enzyme, used in the citric acid cycle, applied to a citrate and water molecule. The reaction is stereospecific, and results therefore in D-isocitrate but not L-isocitrate. In the left side the two central carbon atoms have the TETRAHEDRALSYM shape, in order to match any tetrahedral, while in the right side they both have the more specialized TETRAHEDRALFIXED shape with a specific embedding.



**Fig. 8.** A generic rule that either fixates or changes the embedding of a tetrahedral atom. Each vertex is annotated explicitly with the configuration data, and the asterisks \* are unnamed variable labels that match any atom label.

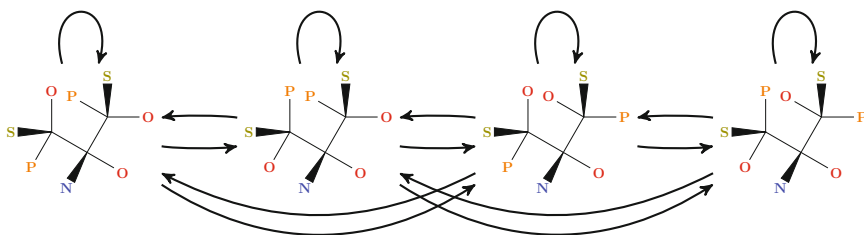
acid without fully specified stereo-information. We see that the 3 stereoisomers, L- and meso-tartaric acid, in addition to the naturally occurring form D-tartaric acid are generated as expected.

While it is not too difficult to manually derive the stereoisomers of tartaric acid, the task quickly becomes complicated and error prone for larger molecules. Enumeration of (i.e., explicitly creating all) and counting molecules has been providing a fertile ground for developments in graph theory, combinatorics, chemistry and the intersecting research fields since the nineteenth century. Many counting problems in chemistry have been solved by the Pólya Theory of



**Fig. 9.** The language of tartaric acid stereoisomers including isomers with generalized stereo configurations, starting from a model without specified tetrahedral embeddings (the graph on top). Each arrow represents a direct derivation using the rule shown in Fig. 8. As it matches any tetrahedral configuration, it also results in identity derivations for molecules already with a TETRAHEDRALFIXED atom. The bottom three graphs, D-, L-, and meso-tartaric acid, are models with fully specified embeddings, and are therefore the proper stereoisomers. The two graphs in the middle only have a tetrahedral embedding fixated on one of the two central carbon atoms, while the other still has TETRAHEDRALSYM shape.

Counting [21,23]. Based on the automorphism group of a molecular graph its cycle index is inferred. The cycle index is used to infer a generating function for which the coefficients correspond to the number of isomers (for an introduction, e.g., see [11]). When applying the theory to stereochemical compounds, considering the order of incident edges of atoms can lead to a non-trivial compensation of stereoisomers (see [22] for an in-depth discussion from a combinatorial point of view). An example is shown in Fig. 10, where a central tetrahedral carbon atom (adjacent to the nitrogen atom) has two graph-isomorphic subtrees attached, i.e., they are isomorphic if the stereo information is ignored. If two different tetrahedral embeddings are added to the subtree carbons, then the central carbon atom can have only one tetrahedral embedding up to isomorphism (the outer graphs of Fig. 10). On the other hand, when two different embeddings are on the subtree carbons, then only two further stereoisomers exist (the inner graphs); one for each of the embeddings on the central carbon. This kind of compensation of such stereoisomers has been thoroughly analysed for specific molecular classes (e.g. tree-like structures with single bonds only) in literature. However, our framework allows not only for enumeration of stereoisomers, but also for a rigorous modelling of chemical and biochemical pathways with complete or partial stereoinformation attached.



**Fig. 10.** The language of all proper stereoisomers for an abbreviated molecule, using the rule shown in Fig. 8. As with the tartaric acid example (Fig. 9) the rule can result in identity derivations. All three carbon atoms have TETRAHEDRALFIXED shape.

## 5 Concluding Remarks

We have presented a model of molecules based on typed attributed graphs that include the representation of local molecular shapes. The model is inspired by previous work on molecule representation, e.g., the ordered list method from chemistry and the hypergraph approach from graph transformation. We have extended it here to allow a partial specification of stereochemical information. This both allows for partially assigning geometric information to molecules, but more importantly provides a more expressive framework for describing classes of reactions as graph transformation rules. The presented model additionally includes the possibility to represent lone electron pairs, which in some cases give rise to multiple stereoisomers. We have implemented the model as an extension of the chemical graph transformation system in the MØD software package. The extension is being prepared for release in an upcoming version of MØD.

*Additional Shapes.* The trigonal planar shape is another important shape in biochemistry, which gives rise to cis-trans isomerism in conjunction with incident double or aromatic bonds. In this shape an atom is coplanar with its required 3 neighbours. In Fig. 4 we have shown how this shape can be added to the shape category. Like the TETRAHEDRALFIXED shape, it has associated constraints on index maps induced by graph morphisms. In addition the trigonal planar shape will also require non-local checks of morphisms to ensure consistency of the half-planes implicitly defined by the neighbour lists.

Shapes that require more than 4 neighbours are uncommon in biochemistry, although the trigonal bipyramid plays a role in phosphorus chemistry. Preliminary investigations suggest that all other chemically relevant local geometries can also be defined in the framework laid out in this contribution.

The embedding of a graph in the plane (or any surface) can be represented by locally imposing a cyclic order on the incident edges at each vertex, also called a rotation system. The semantics of this encoding is similar to that of the trigonal planar shape. The same techniques thus are applicable to defining a transformation system for graphs with an associated embedding.

**Acknowledgements.** This work is supported by the Danish Council for Independent Research, Natural Sciences, the COST Action CM1304 “Emergence and Evolution of Complex Chemical Systems”, and the ELSI Origins Network (EON), which is supported by a grant from the John Templeton Foundation. The opinions expressed in this publication are those of the authors and do not necessarily reflect the views of the John Templeton Foundation.

## A Code Examples

The following code shows how to use the stereochemical extension of MØD, in the context of the three application examples. The code is also available as modifiable scripts in the live version of the software, accessible at <http://mod.imada.sdu.dk/playground.html>.

### A.1 Stereospecific Aconitase

Executing the following code creates the figures for Fig. 7.

```

water = smiles("O", "H_2O")
cit = smiles("C(C(=O)O)C(CC(=O)O)(C(=O)O)O", name="Cit")
d_icit = smiles("C(C([C@@H])([C@H](C(=O)O)O)C(=O)O)C(=O)O", name="D-ICit")

aconitase = ruleGMLString("""rule [
  left [
    # the dehydrated water
    edge [ source 1 target 100 label "-" ] edge [ source 2 target 102 label "-" ]
    # the hydrated water
    edge [ source 200 target 202 label "-" ]
  ]
  context [
    node [ id 1 label "C" ]
    edge [ source 1 target 2 label "-" ] # goes from - to = to -
    node [ id 2 label "C" ]
    # the dehydrated water
    node [ id 100 label "O" ] node [ id 101 label "H" ] node [ id 102 label "H" ]
    edge [ source 100 target 101 label "-" ]
    # the hydrated water
    node [ id 200 label "O" ] node [ id 201 label "H" ] node [ id 202 label "H" ]
    edge [ source 200 target 201 label "-" ]
    # dehydrated C neighbours
    node [ id 1000 label "C" ] node [ id 1010 label "O" ] node [ id 1001 label "C" ]
    edge [ source 1 target 1000 label "-" ] edge [ source 1000 target 1010 label "-" ]
    edge [ source 1 target 1001 label "-" ]
    # hydrated C neighbours
    node [ id 2000 label "C" ] node [ id 2001 label "H" ]
    edge [ source 2 target 2000 label "-" ] edge [ source 2 target 2001 label "-" ]
  ]
  right [
    # The '!' in the end changes it from TetrahedralSym to
    # TetrahedralFixed
    node [ id 1 stereo"tetrahedral[1000, 1001, 202, 2]!" ]
    node [ id 2 stereo"tetrahedral[200, 1, 2000, 2001]!" ]
    # the dehydrated water
    edge [ source 100 target 102 label "-" ]
    # the hydrated water
    edge [ source 1 target 202 label "-" ] edge [ source 2 target 200 label "-" ]
  ]
]""")

dg = dgRuleComp(inputGraphs, addSubset(cit, water) >> aconitase,
  # selcdctino of attributes and morphisms for matching
  labelSettings=LabelSettings(
    # use terms as labels, instead of strings
    LabelType.Term,
    # term morphisms may be specialisations
    LabelRelation.Specialisation,
    # use stereo information,
    # with specialisation in the morphisms
    LabelRelation.Specialisation)
)
dg.calc()
for e in dg.edges:
  p = GraphPrinter()
  p.withColour = True
  e.print(p, matchColour="Maroon")

```

## A.2 Stereoisomers of Tartaric Acid

Executing the following code creates the figures for Figs. 8 and 9.

```

smiles("C(C(=O)O)O(C(=O)O)O", name="Tartaric acid")
smiles("[C@@H]([C@H](C(=O)O)O)(C(=O)O)O", name="L-tartaric acid")
smiles("[C@H]([C@@H](C(=O)O)O)(C(=O)O)O", name="D-tartaric acid")
smiles("[C@@H]([C@@H](C(=O)O)O)(C(=O)O)O", name="Meso-tartaric acid")
change = ruleGMLString("""rule [
  left [ node [ id 0 stereo"tetrahedral" ]
  context [
    node [ id 0 label "*" ] node [ id 1 label "*" ] node [ id 2 label "*" ]
    node [ id 3 label "*" ] node [ id 4 label "*" ]
    edge [ source 0 target 1 label "-" ] edge [ source 0 target 2 label "-" ]
    edge [ source 0 target 3 label "-" ] edge [ source 0 target 4 label "-" ]
  ]
  right [ node [ id 0 stereo"tetrahedral[1, 2, 3, 4]!" ] ]
]""")

dg = dgRuleComp(inputGraphs, addSubset(inputGraphs) >> repeat(change),
  # seldctino of attributes and morphisms for matching
  labelSettings=LabelSettings(
    # use terms as labels, instead of strings
    LabelType.Term,
    # term morphisms may be specialisations
    LabelRelation.Specialisation,
    # use stereo information,
    # with specialisation in the morphisms
    LabelRelation.Specialisation)
)
dg.calc()

p = GraphPrinter()
p.setMolDefault()
p.withPrettyStereo = True
change.print(p)
p = DGPrinter()
p.withRuleName = True
p.withRuleId = False
dg.print(p)

```

## A.3 Non-trivial Stereoisomers

Executing the following code creates the figures for Figs. 8 and 10.

```

g = smiles("[N][C@]([O])([C@]([S])([P])([O]))([C@]([S])([P])([O]))")
change = ruleGMLString("""rule [
  left [ node [ id 0 stereo"tetrahedral" ]
  context [
    node [ id 0 label "*" ] node [ id 1 label "*" ] node [ id 2 label "*" ]
    node [ id 3 label "*" ] node [ id 4 label "*" ]
    edge [ source 0 target 1 label "-" ] edge [ source 0 target 2 label "-" ]
    edge [ source 0 target 3 label "-" ] edge [ source 0 target 4 label "-" ]
  ]
  right [ node [ id 0 stereo"tetrahedral[1, 2, 3, 4]!" ] ]
]""")

dg = dgRuleComp(inputGraphs, addSubset(inputGraphs) >> repeat(change),
  # seldctino of attributes and morphisms for matching
  labelSettings=LabelSettings(
    # use terms as labels, instead of strings
    LabelType.Term,
    # term morphisms may be specialisations
    LabelRelation.Specialisation,
    # use stereo information,
    # with specialisation in the morphisms
    LabelRelation.Specialisation)
)
dg.calc()

p = GraphPrinter()
p.setMolDefault()
p.withPrettyStereo = True
change.print(p)
p = DGPrinter()
p.withRuleName = True
p.withRuleid = False
dg.print(p)

```

## References

1. Akutsu, T.: A new method of computer representation of stereochemistry. Transforming a stereochemical structure into a graph. *J. Chem. Inf. Comput. Sci.* **31**, 414–417 (1991)
2. Andersen, J.L.: MedØIDatschgerl (MØD) (2016). <http://mod.imada.sdu.dk>
3. Andersen, J.L., Flamm, C., Merkle, D., Stadler, P.F.: Inferring chemical reaction patterns using rule composition in graph grammars. *J. Syst. Chem.* **4**(1), 4 (2013)
4. Andersen, J.L., Flamm, C., Merkle, D., Stadler, P.F.: Generic strategies for chemical space exploration. *Int. J. Comput. Biol. Drug Des.* **7**(2/3), 225–258 (2014). <http://arxiv.org/abs/1302.4006>
5. Andersen, J.L., Flamm, C., Merkle, D., Stadler, P.F.: A software package for chemically inspired graph transformation. In: Echahed, R., Minas, M. (eds.) ICGT 2016. LNCS, vol. 9761, pp. 73–88. Springer, Cham (2016). doi:[10.1007/978-3-319-40530-8\\_5](https://doi.org/10.1007/978-3-319-40530-8_5)
6. Benkő, G., Flamm, C., Stadler, P.F.: A graph-based toy model of chemistry. *J. Chem. Inf. Comput. Sci.* **43**, 1085–1093 (2003)
7. Cordella, L., Foggia, P., Sansone, C., Vento, M.: A (sub) graph isomorphism algorithm for matching large graphs. *IEEE Trans. Pattern Anal. Mach. Intell.* **26**(10), 1367 (2004)
8. Cross, L.C., Klyne, W.: Rules for the nomenclature of organic chemistry: section E: stereochemistry. *Pure Appl. Chem.* **45**, 11–30 (1976)
9. Ehrig, H., Ehrig, K., Prange, U., Taentzner, G.: *Fundamentals of Algebraic Graph Transformation*. Springer, Berlin (2006)
10. Ehrig, K., Heckel, R., Lajios, G.: Molecular analysis of metabolic pathway with graph transformation. In: Corradini, A., Ehrig, H., Montanari, U., Ribeiro, L., Rozenberg, G. (eds.) ICGT 2006. LNCS, vol. 4178, pp. 107–121. Springer, Heidelberg (2006). doi:[10.1007/11841883\\_9](https://doi.org/10.1007/11841883_9)
11. Faulon, J.L., Visco Jr., D., Roe, D.: *Enumerating Molecules, Reviews in Computational Chemistry*, vol. 21, pp. 209–286. Wiley, Hoboken (2005)
12. Flack, H.D.: Louis Pasteur’s discovery of molecular chirality and spontaneous resolution in 1848, together with a complete review of his crystallographic and chemical work. *Acta Crystallogr. Sect. A* **65**, 371–389 (2009)
13. Fontana, W., Buss, L.W.: “The arrival of the fittest”: toward a theory of biological organization. *Bull. Math. Biol.* **56**, 1–64 (1994)
14. Fontana, W., Buss, L.W.: What would be conserved “if the tape were played twice”. *Proc. Natl. Acad. Sci. USA* **91**, 757–761 (1994)
15. Gillespie, R.: Fifty years of the VSEPR model. *Coord. Chem. Rev.* **252**, 1315–1327 (2008)
16. Kerber, A., Laue, R., Meringer, M., Rücker, C., Schymanski, E.: *Mathematical Chemistry and Chemoinformatics*. De Gruyter (2013)
17. Kreowski, H.J., Kuske, S.: Graph multiset transformation: a new framework for massively parallel computation inspired by DNA computing. *Nat. Comput.* **10**(2), 961–986 (2011)
18. de Lara, J., Bardohl, R., Ehrig, H., Ehrig, K., Prange, U., Taentzner, G.: Attributed graph transformation with node type inheritance. *Theor. Comput. Sci.* **376**(3), 139–163 (2007). <http://www.sciencedirect.com/science/article/pii/S0304397507000631>
19. Lewis, G.N.: The atom and the molecule. *J. Am. Chem. Soc.* **38**, 762–785 (1916)

20. Petrarca, A.E., Lynch, M.F., Rush, J.E.: A method for generating unique computer structural representation of stereoisomers. *J. Chem. Doc.* **7**, 154–165 (1967)
21. Pólya, G.: Kombinatorische Anzahlbestimmungen für Gruppen, Graphen und chemische Verbindungen. *Acta Mathematica* **68**(1), 145–254 (1937)
22. Pólya, G., Read, R.: *Combinatorial Enumeration of Groups, Graphs, and Chemical Compounds*. Springer, New York (1987)
23. Redfield, J.: The theory of group-reduced distributions. *Am. J. Math.* **49**(3), 433–455 (1927)
24. Rosselló, F., Valiente, G.: Graph transformation in molecular biology. In: Kreowski, H.-J., Montanari, U., Orejas, F., Rozenberg, G., Taentzer, G. (eds.) *Formal Methods in Software and Systems Modeling*. LNCS, vol. 3393, pp. 116–133. Springer, Heidelberg (2005). doi:[10.1007/978-3-540-31847-7\\_7](https://doi.org/10.1007/978-3-540-31847-7_7)
25. Satoh, H., Koshino, H., Funatsu, K., Nakata, T.: Novel canonical coding method for representation of three-dimensional structures. *J. Chem. Inf. Comput. Sci.* **40**, 622–630 (2000)
26. Taentzer, G.: AGG: a graph transformation environment for modeling and validation of software. In: Pfaltz, J.L., Nagl, M., Böhlen, B. (eds.) *Applications of Graph Transformations with Industrial Relevance: Second International Workshop, AGTIVE 2003*. LNCS, vol. 3062, pp. 446–453. Springer, Heidelberg (2004)
27. Weininger, D.: SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Model.* **28**, 31–36 (1988)
28. Wipke, W.T., Dyott, T.M.: Simulation and evaluation of chemical synthesis-computer representation and manipulation of stereochemistry. *J. Am. Chem. Soc.* **96**, 4825–4834 (1974)
29. Yadav, M.K., Kelley, B.P., Silverman, S.M.: The potential of a chemical graph transformation system. In: Ehrig, H., Engels, G., Parisi-Presicce, F., Rozenberg, G. (eds.) *ICGT 2004*. LNCS, vol. 3256, pp. 83–95. Springer, Heidelberg (2004). doi:[10.1007/978-3-540-30203-2\\_8](https://doi.org/10.1007/978-3-540-30203-2_8)