MATCH
Communications in Mathematical and in Computer Chemistry

MATCH Commun. Math. Comput. Chem. **80** (2018) 705-731

ISSN 0340 - 6253

Towards Mechanistic Prediction of Mass Spectra Using Graph Transformation

Jakob L. Andersen[1,2,10,11], **Rolf Fagerberg**[1], **Christoph Flamm**[2,9], **Rojin Kianian**[1,3,4], **Daniel Merkle**[1,*], **Peter F. Stadler**[2–8]

[1] *Department of Mathematics and Computer Science, University of Southern Denmark, Odense M DK-5230, Denmark*

[2] *Institute for Theoretical Chemistry, University of Vienna, Wien A-1090, Austria*

[3] *Bioinformatics Group, Department of Computer Science, University of Leipzig, Leipzig D-04107, Germany*

[4] *Max Planck Institute for Mathematics in the Sciences, Leipzig D-04103, Germany*

[5] *Interdisciplinary Center for Bioinformatics, and German Centre for Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig and Competence Center for Scalable Data Services and Solutions Dresden-Leipzig and Leipzig Research Center for Civilization Diseases, University of Leipzig, Leipzig D-04107, Germany*

[6] *Fraunhofer Institute for Cell Therapy and Immunology, Leipzig D-04103, Germany*

[7] *Center for non-coding RNA in Technology and Health, University of Copenhagen, Frederiksberg C DK-1870, Denmark*

[8] *Santa Fe Institute, 1399 Hyde Park Rd, Santa Fe NM 87501, USA*

[9] *Vienna Metabolomics Center (VIME), University of Vienna, Wien A-1090, Austria*

[10] *Earth-Life Science Institute, Tokyo Institute of Technology, Tokyo 152-8550, Japan*

[11] *Research group Bioinformatics and Computational Biology, Faculty of Computer Science, University of Vienna, Wien A-1090, Austria*

∗ daniel@imada.sdu.dk

(Received June 9, 2017)

Abstract

We suggest a line of work for improving the current state-of-the art in computational methods for mass spectrometry. Our main focus is on increasing the chemical realism of the modeling of the fragmentation process. Two core ingredients of our proposal are i) describing the individual fragmentation reactions via graph transformation rules and ii) expressing the dynamics of the system via reaction rates and quasi-equilibrium theory. We use graph transformation rules both for specifying the possible core fragmentation reactions, and for characterizing the reaction sites when learning values for the rates. We employ a strategy framework in order to systematically expand the chemical space of fragments. We think that this approach in terms of chemical modeling is more mechanistically explicit than previous ones, and believe this can lead to both better spectrum prediction and more explanatory power. Our modeling of system dynamics also allows better separation of instrument dependent and instrument independent parameters of the model.

1 Introduction

Mass spectrometry is an analytic technique for characterizing single molecules and molecular mixtures which uses the fact that acceleration of charged particles in an electric field depends on their mass-to-charge ratio. Ionization of the sample molecules transfers sufficient energy to trigger their fragmentation. The outcome of such experiments are mass spectra containing peaks representing the (relative) abundances of fragments as a function of their mass-to-charge ratio. For an example of a spectrum, see Fig. 3 with m/z (mass m, charge z) as abscissa and intensity (abundances of fragments) as ordinate. The pattern of fragment masses and abundances contains ample information on the molecular structure. While it is sufficient in most cases to identify a known molecule, it has remained an excruciatingly difficult and often impossible task to determine the structure of an unknown molecule by mass spectrometry alone [1].

The mass spectrometry-based identification of small molecules usually relies on the search of the corresponding mass spectrum in a reference spectrum database [2–5]. This approach, however, will not be successful for molecules that do not yet have a database entry, and therefore critically depends on computational methods for the *in silico* prediction of a mass spectrum from the molecular structure [6]. Four very different computational approaches are widely used throughout the literature: (i) rule-based fragmentation (ii) combinatorial fragmentation (iii) fragmentation trees, and (iv) competitive fragmentation. The first three approaches aim at ranking a set of candidate structures by their potential to explain a maximum number of informative peaks that are observed in the mass spectrum of an unknown compound. Peaks with high m/z values, and among those the peaks with higher intensities, usually convey more structural information than peaks with low m/z values and are deemed informative peaks. A peak in a mass spectrum is explained by a candidate structure if a fragment with corresponding mass, within user-defined error bounds, can be generated via bond breaking processes. The candidate structure which explains the maximum number of informative peaks is the most likely to have generated the unknown mass spectrum. Being based on the number of matched peaks, none of these approaches take the full intensity information in the mass spectrum into account. A mass spectrum without intensity information is frequently called a barcode spectrum. The most popular computational tools for these three approaches are the open source software MetFrag [7, 8], the commercial packages Mass Frontier and MOLGEN-MSF [9], and the package SIRIUS [10] for which no source code is available. The methods differ mainly in the way the fragmentation graphs are constructed. While Mass Frontier and MOLGEN-MSF rely on a large, hand-curated set of experimentally observed fragmentation reactions, MetFrac systematically breaks every bond in the candidate molecule and prunes this highly combinatorial search space with heuristic filtering strategies based on physicochemical properties of the broken bonds and generated fragments. MOLGEN-MSF in addition calculates the theoretical isotope fine structure for each predicted fragment and filters out fragments where the predicted isotope fine structure does not match the isotope fine structure of the targeted peak in the reference spectrum. The key concept of SIRIUS is the so-called fragmentation tree [11], which explains the fragmentation cascade on the level of the molecular formulas. The fragmentation tree is directly calculated from the mass spectral data using combinatorial optimization techniques; this fact sets SIRIUS apart from the other methods.

Competitive Fragmentation Modeling (CFM) [12–15] is among the current state-of-the-art methods and the only one that tries to predict mass spectra including peak intensities from molecular structures. The method combines a restricted combinatorial fragmentation method with a Markov-type dynamics on the generated fragmentation graph to estimate intensity values for the predicted fragments. Parameters such as bond breaking propensities are inferred from experimental mass spectrum data with the help of machine-learning techniques. The idea was pioneered already in [16] for a specific class of fragmentation reactions. The computational model parameters can be trained for different experimental techniques (e.g. hard and soft ionization methods), which makes this approach quite flexible. The source code as well as a web service CFM-ID [12] are available. For recent reviews on computational approaches in small molecule mass spectrometry we refer to [6, 17].

Semiempirical methods have successfully been applied to calculate bond orders, energy partitioning, or ionization potentials for small molecular systems (see literature reviewed in [18]). However, there are currently no parameters available for third-row elements, and Silicium for instance is an integral part of derivatization agents for gas chromatography-mass spectrometry approaches. Furthermore, to obtain reliable estimates for the activation energy of bond fission, a high level of theory is required, which makes practical computations very expensive. Recently, first-principle methods from quantum mechanics have been applied to the *de-novo* prediction of electron ionization mass spectra [18–20]. These methods are, however, computationally extremely expensive. At least at the present they are not suited for high-throughput applications.

In this contribution we describe a road map for improving the current computational methods for the prediction of mass spectra. In our view, the key issue is achieving a chemically more realistic modeling of the fragmentation process without sacrificing computational feasibility. To this end, we propose to (i) describe the individual fragmentation reactions by means of graph transformation rules and (ii) to express the dynamics of the system in terms of reaction rates and quasi-equilibrium theory. Graph transformation rules strike a useful balance between chemical expressiveness and computational efficiency. We use them here both to specify the possible core fragmentation reactions, and to characterize the reaction sites when learning values for the rates. We argue that the mechanistically more explicit modeling of the underlying chemical reality not only holds more explanatory power but also promises substantial practical improvements for the prediction of mass spectra. Missing lines in predicted spectra, for instance, are directly indicative that fragmentation reactions are missing from the rule set, which the mechanistic nature of the model makes much easier to identify and correct. Our modeling of system dynamics also allows better separation of instrument dependent and instrument independent parameters of the model.

This paper is organized as follows: In Section 2, we define the formal model for describing graph transformation rules and strategy frameworks. In Section 3, we give an overview of how fragmentation graphs are used to predict mass spectra. In Section 4, we explain how graph transformation can be used to find the fragmentation graphs of molecules. In Section 5, we explain our modeling of systems dynamics and outline how the rates and other parameters can be learned from mass spectrometry data and then used to predict spectra of further molecules. The core of our proposal is embodied by Sec. 4 and 5.

2 Graph Transformation

Any computational approach for predicting mass spectra inherently requires a representation of molecules and a method for transforming molecules via fragmentation reactions. This essentially amounts to a model for an artificial chemistry. Several abstraction approaches exist for molecules (e.g., the molecular formula, the structural formula, and approaches that include partial or complete information on the three-dimensional arrangement of atoms) as well as for chemical reactions (e.g., the λ-calculus [21], the chemical abstract machine [22], or graph transformation systems [23]). In this contribution we model chemical compounds as labeled undirected graphs and chemical reactions as graph transformation rules utilizing the Double Pushout (DPO) approach [24]. In a nutshell, graph transformation rules add reaction modeling to the classic modeling of molecules as static graphs. We note that all the formal modeling described in this section has been implemented in an efficient software library MØD [25].

2.1 Graph Transformation Framework

Chemical reactions usually act on more than one compound and create more than one compound. Therefore we consider graph transformation that in general can operate on graphs whose connected components represent the individual molecules. As educts and products of reactions can appear in multiplicities, we will implicitly interpret a set of connected graphs as a multiset of connected graphs when applying a rule. A rule p thus transforms a multiset of connected graphs (usually denoted G) to a multiset of connection graphs (usually denoted H). A DPO transformation rule $p = (L \xleftarrow{l} K \xrightarrow{r} R)$ formally consists of three graphs L, R, and K, as well as two graph morphisms l and r. Such a rule can be applied to G if the graph L can be found as a subgraph in G, i.e., if a matching graph morphism $m\colon L \to G$ exists. The copy of L in G is then replaced by R by first removing $L \backslash K$ from G, obtaining an intermediary graph D, to which $R \backslash K$ is added. The result is the new graph H, which again may consist of multiple connected components.[1] Such a direct derivation is denoted $G \overset{p,m}{\Longrightarrow} H$, or simply $G \overset{p}{\Rightarrow} H$ or $G \Rightarrow H$ when the matching morphism or rule is not relevant. By splitting H into its connected components we obtain a multiset of connected graphs that correspond to the products of a chemical

[1] Formally, the two squares, see Fig. 1, are *pushouts*, a concept form category theory. This means G is the gluing of L together with D along the common graph K, and similarly for H with R, D, and K For brevity, we here skip defining this constraint—see [24] for full details.

reaction. For a formal and detailed introduction and a comparison with other modeling approaches we refer the reader to [23, 24, 26].

An example of a rule p and its application is depicted in Fig. 1.

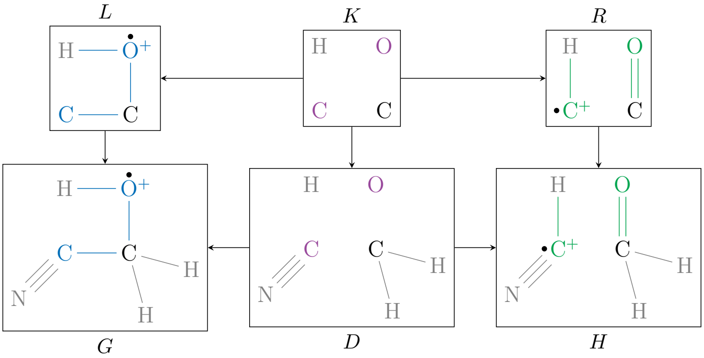


Figure 1. A direct derivation in the Double Pushout approach to graph transformation. The transformation rule $p = (L \overset{l}{\Leftarrow} K \overset{r}{\Rightarrow} R)$ is applied to the graph G by finding a subgraph match of L in G, i.e., a match morphism $m\colon L \to G$. The resulting graph H is derived by first removing $L\backslash K$ from G, and then adding $R\backslash K$. The rule p represents H-Y elimination.

The top span corresponds to the transformation rule p and the bottom span corresponds to the application of p to G leading to H. The reaction mechanism encoded by the rule in Fig. 1 is commonly known as H-Y elimination, which splits a compound in a mass spectrometer. We underline that graph transformation facilitate the construction of rules that are chemical in nature, i.e., there is a chemical reaction underlying the transformation rule. Also note that the graph L of a rule does not need to match entire molecules, but merely parts of them. In other words, L defines a neighborhood of the reaction site, which specifies exactly where in molecules this rule can be applied. Hence, a single rule can capture the same type of reaction in many molecules, and even in several places in the same molecule.

2.2 Strategy Framework

For the prediction of a mass spectrum, and for compound identification when spectra are given, the chemical space of potential fragments needs to be computed. Computational approaches for this often suffer from a combinatorial explosion. For example, in order to keep the computational cost within tractable limits, only two consecutive fragmentation steps were considered for applications of CFM-ID reported in [12].[2] In [27], the strategic graph transformation framework was presented, which allows the use of sophisticated strategies for expanding the chemical space of interest more systematically. In the following we present an abbreviated version of the framework, as needed for this contribution.

The core of the framework is rule application. Given a set of connected graphs U and a graph transformation rule p, the application $p(U)$ of p on U is a new set of connected graphs U' defined as follows:

$$U' = U \cup \{h \mid \exists G \subseteq U : G \overset{p}{\Rightarrow} H \text{ and } h \in \mathrm{CC}(H)\} \tag{1}$$

That is, for each G on which p can be applied, we add the set of connected components $\mathrm{CC}(H)$ of the H resulting from the rule application. Repeated application of a rule can then accumulate a set of reachable graphs from the initial set. In a chemical context, this amounts to the construction of a reaction network.

The strategy framework is a domain specific programming language for graph transformation, in which rule application is the fundamental operation. In the framework, a strategy is a function from a set of graphs to another set of graphs. A single rule p is a strategy by itself, whose application on a set of graphs is as defined above. Existing strategies can be composed to new strategies in various ways, the semantics of which we now briefly describe. A full description of the framework can be found in [27].

As above, we use U to denote the set of input graphs, and U' the resulting set of graphs. The letter Q is used to denote arbitrary strategies, and square brackets are used to delimit parameters of strategies. The notational scheme for the application of a strategy is thus $U' = \texttt{strat}[\mathit{parameter}](U)$.

Parallel. Given a set of strategies $\{Q_1, Q_2, \dots, Q_n\}$, the strategy $\texttt{parallel}[\{Q_1, Q_2, \dots, Q_n\}]$ applies each Q_i to the input independently, and returns the union of the results.

Sequence. Given strategies $Q_1, Q_2, \dots, Q_n$, we write $Q = Q_1 \rightarrow Q_2 \rightarrow \cdots \rightarrow Q_n$ for the composition of applying each strategy, i.e., $Q(U) = Q_n(\dots(Q_2(Q_1(U)))\dots)$. Additionally, $Q = Q'^n$ denotes the n-fold composition of the strategy Q'.

[2] In [15], Allen notes that experiments with a search depth of three were carried out, but the increased computational effort did not substantially improve the results.

Repeat. Given a strategy Q' and a non-negative integer n, the strategy $\texttt{repeat}[n, Q']$ is equivalent to Q'^k for the smallest k such that either $n = k$, or the graph set is empty, or the graph set has reached a fixed point.

Filter. Given a predicate P on graphs, a filter strategy $\texttt{filter}[P]$ returns the set of graphs satisfying the predicate.

Derivation Predicate. Given a predicate P on direct derivations and a strategy Q', we write $\texttt{derivationPredicate}[P, Q']$ for a derivation predicate strategy. It executes Q' on the input, but modifies rule application, Eq. (1), so only direct derivations satisfying the predicate are accepted.

In short, reaction networks can be built from one or more starting molecules (graphs) using repeated application of rules, and the strategy framework offers a flexible way of specifying and controlling the construction of such networks.

3 Overall Model for Mass Spectrum Prediction

The overall model we use for mass spectrum prediction is based on the concept of *fragmentation graphs*. A single fragmentation breaks a charged molecule into two parts, of which one receives the charge and the other stays neutral. As only charged molecules are detectable in a mass spectrometer, only the charged fragment is relevant, hence each fragmentation can be considered a one-to-one reaction. This charged fragment may then fragment again into even smaller fragments. Therefore, the possible ways of repeatedly fragmenting a charged molecule M into smaller parts is well represented by a directed acyclic graph (DAG), denoted the fragmentation graph for M. The nodes of the DAG are the possible fragments, the arcs of the DAG are the one-to-one fragmentation reactions, and M is the source of the DAG. One may include the ionization step in the DAG by allowing the source to be the uncharged version of M, and letting every outgoing arc of the source point to an ionized version of M. Each outgoing arc therefore corresponds to an independent reaction channel in the language of chemical kinetics.

As an example consider Fig. 2, depicting a reaction network for the fragmentation of the molecule glycolonitrile (CAS 107-16-4). The fragmentation graph is marked as a blue subgraph and everything else are neutral fragments which are not detectable. Each

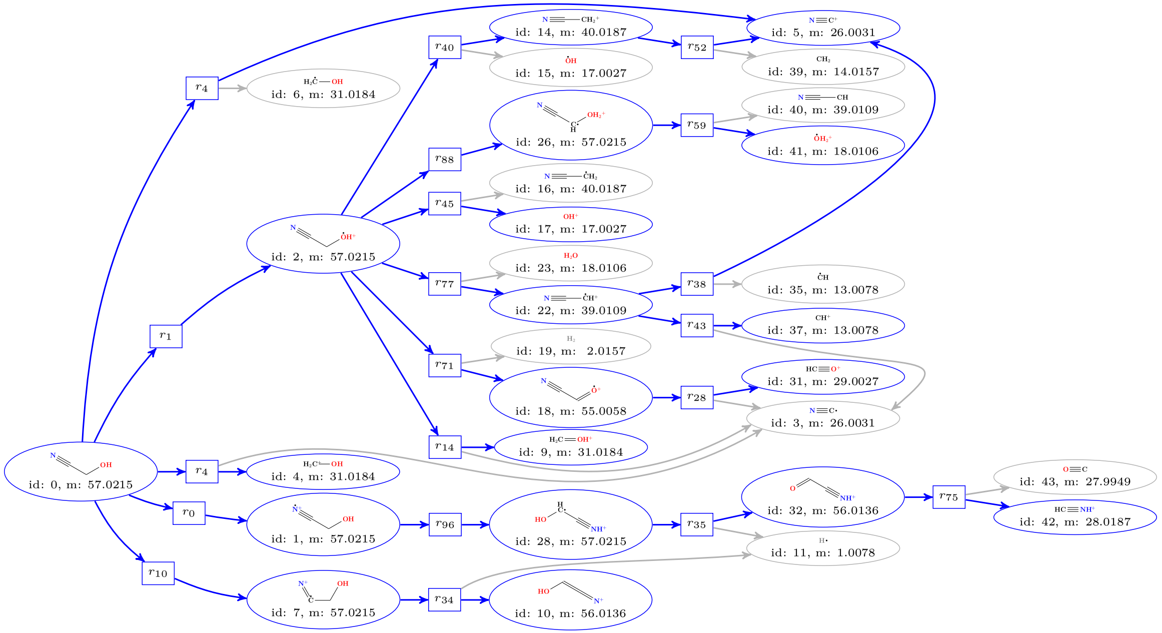


Figure 2. Reaction network for the molecule glycolonitrile (CAS 107-16-4).
The fragmentation graph, highlighted in blue, is a directed acyclic graph of the charged molecules observable by mass spectrometry. The five arcs leaving the leftmost vertex (glycolonitrile, id 0) correspond to ionization processes as they typically occur in the ionization method Electron Ionization (EI). The reference spectrum for glycolonitrile (NIST MS number 230418) is depicted in Fig. 3.

fragment is shown with an id and its monoisotopic mass. Each reaction also has an identifier, whose precise meaning will become clear in Sec. 4. Though glycolonitrile is a small molecule, it gives rise to several different types of fragmentations, including a rearrangement (r_{96}) and three elimination (r_{71}, r_{75}, r_{77}) reactions. It also illustrates that ionization (r_0, r_1, r_4, r_{10}) can yield multiple different ions, here five, as during EI electrons from different atom/bonds of the molecule can be removed to form the primary molecule ion M^+. This ion is thus a mixture of different ionization states rather than a single homogeneous ion. Note that the compound cyanide (id 5) can be obtained in three ways, both as a product of cleavage (r_{38} and r_{52}) but also as a consequence of ionization (r_4).

If the probability of each fragmentation in a fragmentation graph is known, the graph translates into a spectrum: For any given fragment, a path of consecutive fragmentations leading to it has a probability given by the product of the probabilities of the fragmentations along it; the sum over all such paths gives the intensity at which the fragment occurs. Different ions with the same mass and charge will all contribute to the corresponding peak in the mass spectrum. For example, in Fig. 2 this means that the five ions with monoisotopic mass 57.02 u (nodes with id 1, 2, 7, 26 and 28) will all contribute to the peak at m/z 57.02 (identified as glycolonitrile ion).

Thus, the overall model we consider has three phases:

1. Generation of the fragmentation graph
2. Estimation of probabilities of the fragmentations
3. Construction of a spectrum.

This overall model has been used before, most closely related to our line of work by Allen et al. in a series of works [12–15] leading to the program CFM-ID [28]. For comparison with our proposal below, we outline their approach: For phase 1, they systematically break all non-hydrogen bonds, each time ensuring satisfaction of a set of chemically motivated constraints (such as mass preservation, charge preservation, and possible hydrogen rearrangements) by using integer linear programming. For phase 2, they introduce a break tendency for each fragmentation, i.e., how likely it is that a fragmentation will occur. The break tendencies are learned using machine learning, using models built on the assumption that similar molecules break in similar ways. To quantify similarity, each fragmentation is associated with a feature vector, a binary vector in which each entry i indicates if the fragmentation has feature i. Most features are characteristics of the surroundings of the bond broken by the fragmentation, e.g., the presence of particular atoms within a certain distance or if a broken ring was aromatic or not. Using this feature vector as input, they train two types of models for the break tendency, a linear model and a neural network. Finally, each break tendency is translated into a probability for the fragmentation using a softmax function [29] taking into account the other possible fragmentations for the same molecule, i.e., other outgoing arcs in the fragmentation graph.

The core of our proposal is performing phase 1 and phase 2 using new approaches. This is the subject of Sec. 4 and 5, respectively. Phase 3 is a fairly straightforward calculation based on the fragmentation graph and the probabilities as described above, and will not be discussed in more detail here.

4 Using Graph Transformation for Modeling Fragmentation

Graph transformation is a formalism well suited for the generation of fragmentation graphs, since it allows precise and flexible specification of fragmentation reactions, based on knowledge of the underlying chemistry. Given a set of appropriate graph transformation rules and a derivation strategy, generation of the fragmentation graph is a straightforward application of these to the input molecule. In this section we illustrate by an example how to use graph transformation rules and the strategy framework for the construction of fragmentation graphs.

Consider once again the graph transformation rule depicted in Fig. 1. This models a rule used for fragmentation, namely an H-Y elimination. It is part of the rule set used to create the fragmentation graph in Fig. 2. The other rules used for this graph are listed in App. A. All are part of a database of ionization and fragmentation rules, which we are currently building. The rules above were extracted manually from McLafferty's book [30]. Now, consider the reaction network depicted in Fig. 2, and recall that the fragmentation graph is the blue subgraph. The network is produced by first applying the ionization rules to the input graph, with subsequent removal of charge neutral fragments. Then, the fragmentation rules are applied repeatedly (here four times) to the remaining charged fragments, also with removal of neutral fragments in each iteration. Note, that the number of fragmentation steps is configurable, which can reflect e.g. different ionization methods.

Thus, the strategy is as follows.

$$
\begin{aligned}
&\texttt{parallel}[R_{\text{ionization}}] \rightarrow \texttt{filter}[P_{\text{charge}}] \rightarrow \texttt{repeat}[4,\\
&\qquad \texttt{parallel}[R_{\text{fragmentation}}] \rightarrow \texttt{filter}[P_{\text{charge}}]\\
&]
\end{aligned}
$$

with

$$P_{\text{charge}}(g) = \text{charge}(g) \neq 0$$

Using the software package MØD [23, 25] the strategy above is implemented in the following way:

```
targetCompounds = [smiles("N#CCO")]

def hasCharge(g, gs, first):
   return sum(v.charge for v in g.vertices) != 0

strat = (
   ionizationRules
   >> filterSubset(hasCharge)
   >> repeat[4](
      fragmentationRules >> filterSubset(hasCharge)
   )
)
dg = dgRuleComp(inputGraphs, addSubset(targetCompounds) >> strat)
dg.calc()
dg.print()
```

The strategy framework can also call external programs in order to control the expansion process. For instance, if we want to only apply fragmentation rules to the charged fragments that are most likely to fragment further, an external program can be called for estimating such probabilities for the different fragments and reactions. Examples of other, more advanced strategies are found in App. C. Naturally, different strategies lead to different fragmentation graphs.

To give a better understanding of our model and its advantages, we will look a bit closer at the fragmentation graph for glycolonitrile in Fig. 2 and compare it to the reference spectrum for glycolonitrile depicted in Fig. 3 (NIST MS number 230418).

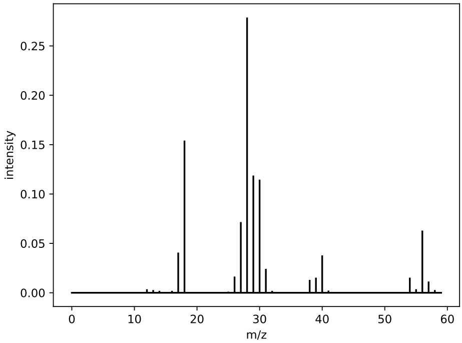


Figure 3. The EI spectrum of glyconitrile (NIST MS number 230418).

The spectrum is a low-resolution spectrum, hence m/z values are treated as integers. The two most abundant peaks in the spectrum are located at m/z 18 and m/z 28. There are two vertices in the fragmentation graph with m/z 18, namely vertices 41 and 23, both water, but only vertex 41 is ionized and therefore detectable by the spectrometer. The peak at m/z 28 can be explained by vertex 42 in the fragmentation graph. The two peaks are generated from different M^+ ionization states, both of which are included in the fragmentation graph. While peak m/z 28 (vertex 42) is produced from M^+ ionized at the nitrogen atom, via a McLafferty-type Hydrogen-rearrangement and loss of carbon-monoxide reaction cascade, peak m/z 18 (water, vertex 41) is generated from M^+ ionized at the oxygen atom via a reaction sequence typical for primary alcohols.

Another abundant peak in the reference spectrum, Fig. 3, is at m/z 30, a peak with no corresponding vertex in Fig. 2. The missing vertex is a sure indication that the model is incomplete, because it means that no predicted spectrum based on this fragmentation graph will include the peak. This implies that chemical rules are missing. We will use this example to illustrate how to develop a rule set. Taking a closer look at the fragments in the fragmentation graph, note that there is a fragment, vertex 9, with m/z 31. It is likely that the hydrogen of the hydroxyl group of this fragment will split off, leaving the charge behind, which would leave a fragment with m/z 30. Adding a rule describing this mechanism to the rule set would yield the missing peak in the predicted spectrum. The rule is left out for illustration purposes. We stress that this way of improving the model is a feature of the mechanistic approach. In contrast, the fragmentation graph computed by CFM-ID (depicted in App. B) also has a fragment missing compared to the reference spectrum in Fig. 3, namely for the peak at m/z 18 (water), but the method of CFM-ID does not facilitate an easy way to develop the fragmentation graph based on this observation.

With a physical modeling approach of the fragmentation process, graph transformation rules may also be automatically inferred. We suggest a strategy to infer a missing fragmentation reaction (or a missing vertex in the fragmentation network) which combines Böcker's [11] fragmentation tree approach with isomer generation [31] and reaction perception [32]. The fragmentation tree approach [11] makes it possible at the level of molecular formulas to determine which neutral fragment has been split off from which vertex. While the structure of the parental compound is known, the structure of the neutral fragment remains undetermined. However, structures may be generated from the inferred molecular formulas for the neutral and the charged fragments using a structural isomer generator such as MOLGEN 5.0 [31]. Finally, reaction perception is performed for all possible one-to-two combinations of the parental structure and the inferred fragment structures to perceive a possible reaction mechanism for the missing fragmentation reaction. It is important to note that complicated rearrangement mechanisms can also be found by this strategy. It should therefore be possible to extract specific fragmentation mechanisms automatically from mass spectral data.

Other advantages of using graph transformation rules for computer-aided mass spectrometry are the various ways one can customize the rule set used. For instance, the rule set may be altered to fit specific molecular classes and to specific experimental setups (Electron Impact Ionization, Electrospray Ionization, positive ions, negative ions). In contrast to methods for generation of fragmentation graphs proposed earlier, the use of graph transformation also makes is possible to trace specific atoms through the fragmentations, important for instance in analysis techniques for biological networks based on isotope-labeling.

5 Better Modeling of Fragmentation Dynamics

In line with the arguments presented by Gasteiger et al. in [16] we are convinced that only a full mechanistic modeling of the various processes which organic molecules undergo in the mass spectrometer will result in major advances in the computational prediction of mass spectra from the molecular structure. To this end, we propose to use a chemical model also for the calculation of fragmentation probabilities in phase 2, instead of relying on a "black box" machine learning approach. In the following we outline how this can be achieved.

The fate of a molecular ion in the mass spectrometer is governed by two broad classes of reactions: (i) fragmentation reactions, where the ion splits up via bond breaking processes into two or more smaller parts, and (ii) rearrangement reactions, where the ion's skeleton is reconfigured to a structural isomer of the original ion. Both reaction classes are instances of unimolecular reactions, which means that the rate of a particular reaction depends only linearly on the reactant. The dynamics of such reaction systems is usually modeled by systems of linear ordinary differential equations of the following form

$$\frac{d}{dt}p(t) = R \cdot p(t)\,,$$

where R is an $n \times n$ rate matrix. Each entry r_{ij} represents the transition rate from state i to state j, i.e., the rate with which compound i fragments into compound j. The vector $p(t)$ holds the population density in the n states of the system for time t. The formal solution of the above matrix equation

$$p(t) = \exp\{t \cdot R\} \cdot p(0)$$

requires an efficient computation of the matrix exponential [33]. Assuming that the topology of the fragmentation graph and the reaction rate structure on top of it correctly represents the real situation in the mass spectrometer, then the recorded mass spectrum is $p(\tau)$, where τ corresponds to the time available for the fragmentation reaction. In practise, the best we can hope for is that $p(\tau)$ approximates the measured spectrum. In order to estimate the τ, we can conveniently use the decay of the M^+-peak of the original molecular ion. That is, we ask for which value of τ the simulated value $p_{\mathrm{M}^+}(\tau)$ agrees with the relative intensity of the observed M^+-peak.

The rate r_{ij} for the transition from state j to state i can be written according to the quasi-equilibrium theory (QET) [34–36] as

$$r_{ij} = A \cdot \exp\{\Delta G^{\ddagger}/E_a\}$$

The Arrhenius-type reaction rate r_{ij} depends on the quotient between the free energy $\Delta G^{\ddagger}$ of the transition state and the ionization energy E_a (A is an entropic factor which is usually set to 1). The abundances of fragments in the spectrum is a function of the ionization energy. This relation between fragment abundance and ionization energy can be measured experimentally [37] and is referred to as a breakdown graph in the mass spectrometry literature. Breakdown graphs thus can be simulated directly with this approach.

The break tendencies of the same bond type in different molecular contexts will for sure differ due to the impact of the specifics of the bond's local surrounding (e.g., steric hindrance, ring context, etc.). Instead of learning the parameters $\Delta G^{\ddagger}$ directly from measured mass spectra of known structures, the idea of local descriptors can be used to express the $\Delta G^{\ddagger}$ values in terms of sets of empirically derived additive contributions that depend explicitly on the bonds' local neighborhoods. We note that the matching mechanism of graph transformation rules (the graph L, cf. Fig. 1) is an ideal vehicle for characterizing such local neighborhoods. For example, the contributions to a single bond between two carbon atoms may be written in the following form:

$$\Delta G^{\ddagger}_{\mathrm{C\text{-}C}} = c_1 \cdot g\left(\mathrm{C\text{-}C}\right) + c_2 \cdot g\left({}_{\diagup}\mathrm{C\text{-}C}^{\diagup}\right) + c_3 \cdot g\left({}^{\diagdown}_{\diagup}\mathrm{C\text{-}C}^{\diagup}\right) + c_4 \cdot g\left({}^{\diagdown}_{\diagup}\mathrm{C\text{-}C}^{\diagup}_{\diagdown}\right) + \ldots$$

Group-contribution methods [38], which express the physicochemical property of interest as a function of structure-dependent parameters, are widely used in Chemistry because experimentally determined values only are available for a small subset of the known compounds. Group-contribution methods make it possible to estimate values for properties such as melting and boiling points, or standard Gibbs energies, without the need for massive computational resources. Thus, they can be applied to process large numbers of molecules. Parameters are usually determined by regression methods from an empirical data set. To achieve higher accuracy, the data set used for learning can be restricted to a particular class of molecules such as fatty acids. Recently this approach has been applied to estimate standard Gibbs energies of biochemical reactions [39] and to evaluate the thermodynamic and kinetic quality of different pathway chemistries that produce the same molecules [40].

Missing lines in a predicted mass spectrum can have two causes: (i) the fragmentation chemistry does not generate a corresponding charged fragment at all, or (ii) the rate for the reaction channel producing the unobserved charged fragments is to small. While the first cause is a missing reaction mechanism, i.e., a missing graph transformation rule, as described in Sec. 4, the latter is due to incorrect parameters inferred in the machine learning step. The underlying assumption is that if the fragmentation network is mechanistically correct, it must be possible to infer correct reaction rates for the different reaction channels from mass spectral data. For a single mass spectrum, the inference of reaction rates is usually under-specified. However, for a series of mass spectra, where the same fragmentation reactions occur in different molecular contexts, there are enough constraints from the peak intensities to make the inference of reaction rates a well-defined problem. Furthermore, the field of machine learning has recently experienced a tremendous advance with the development of deep learning approaches [41], which has yet to be applied in the context of mass spectrometry.

In addition to providing a chemical description of the various intensities in a predicted spectrum, the model naturally includes and clearly separates spectrometer-specific parameters such as E_a and A from parameters referring to molecular structures. This makes it possible to train a model on one spectrometer and transfer it to another. For instance, the explicit representation of the ionization energy E_a renders training for multiple energy levels unnecessary. A model may be trained for one value for E_a and used to predict spectra for a different energy level, or used to learn the energy level for a different spectrometer.

6 Conclusion

We have described a road map for improving the current computational methods for prediction of mass spectra to a chemically more realistic model of the fragmentation process. The overall model has three phases: (1) generation of the fragmentation graph, (2) estimation of probabilities of the fragmentations, and (3) construction of a spectrum.

For phase one, graph transformation strikes a useful balance between chemical expressiveness and computational efficiency. We have demonstrated how to model individual ionization- and fragmentation reactions as graph transformation rules, which may be used both to specify the possible core fragmentation reactions, and to characterize the reaction sites when learning values for the rates. We have also described how to use the strategy framework for systematically building fragmentation graphs, ensuring computational efficiency by limiting the combinatorial explosion of the chemical space of candidate fragments. The mechanistically more explicit model of the underlying chemical reality not only holds more explanatory power but also promises substantial practical improvements for the prediction of mass spectra. Missing lines in predicted spectra, for instance, are directly indicative of an incomplete model, which the mechanistic nature of our approach makes much easier to identify and correct, by adding rules to the rule set or adjusting the strategy. The rule set may also be altered to fit specific machines, facilitate both Electron Impact Ionization and Electrospray Ionization, and positive and negative charges. In contrast to the previously proposed methods for generation of fragmentation graphs, the use of graph transformation also makes is possible to trace specific atoms through the fragmentations.

For phase two, we proposed using a chemical model inspired by the physical processes of fragmentation for learning rates, because we believe that black box machine learning approaches will not be key in major advances of computational prediction of mass spectra. The model is based on QET and parametrized using an empirical increment system on graph descriptors, and in contrast to previous models it has a clear separation of spectrometer-dependent and -independent parameters. This makes it possible to use the same mechanistic model in multiple labs—trained to fit each specific spectrometer. It also renders training for multiple energy levels unnecessary. Furthermore, the natural inclusion of the ionization energy makes it possible to simulate breakdown graphs.

The work presented here is one step towards improving the prediction of mass spectra. Our emphasis is not only to compute accurate spectra, but also to infer meaningful descriptions of the chemical processes taking place in a mass spectrometer.

Acknowledgments: This work was supported in part by the Volkswagen Stiftung proj. no. I/82719, the COST-Action CM1304 “Systems Chemistry”, by the Danish Council for Independent Research, Natural Sciences, grants DFF-1323-00247 and DFF-7014-00041, by the National Science Foundation (INSPIRE 1648973), and by 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.

Bibliography

[1] K. Scheubert, F. Hufsky, S. Böcker, Computational mass spectrometry for small molecules, *J. Cheminformatics* **5** (2013) #12.

[2] L. Sumner, A. Amberg, D. Barrett, M. Beale, R. Beger, C. Daykin, T. Fan, O. Fiehn, R. Goodacre, J. Griffin, T. Hankemeier, N. Hardy, J. Harnly, R. Higashi, J. Kopka, A. Lane, J. Lindon, P. Marriott, A. Nicholls, M. Reily, J. Thaden, M. Viant, Proposed minimum reporting standards for chemical analysis, *Metabolomics* **3** (2007) 211–221.

[3] B. L. Milman, General principles of identification by mass spectrometry, *Trends Anal. Chem.* **69** (2015) 24–33.

[4] S. Böcker, Searching molecular structure databases using tandem MS data: are we there yet? *Curr. Opin. Chem. Biol.* **36** (2017) 1–6.

[5] T. Kind, H. Tsugawa, T. Cajka, Y. Ma, Z. Lai, S. S. Mehta, G. Wohlgemuth, D. K. Barupal, M. R. Showalter, M. Arita, O. Fiehn, Identification of small molecules using accurate mass MS/MS search, *Mass Spectrom. Rev.* **9999** (2017) 1–20.

[6] F. Hufsky, S. Böcker, Mining molecular structure databases: Identification of small molecules based on fragmentation mass spectrometry data, *Mass Spectrom. Rev.* **9999** (2016) 1–10.

[7] C. Ruttkies, E. L. Schymanski, S. Wolf, J. Hollender, S. Neumann, MetFrag relaunched: incorporating strategies beyond *in silico* fragmentation, *J. Cheminf.* **8** (2016) #3.

[8] S. Wolf, S. Schmidt, M. Müller-Hannemann, S. Neumann, In silico fragmentation for computer assisted identification of metabolite mass spectra, *BMC Bioinf.* **11** (2010) #148.

[9] A. Kerber, M. Meringer, C. Rücker, CASE via MS: Ranking structure candidates by mass spectra, *Croat. Chem. Acta* **79** (2006) 449–464.

[10] S. Böcker, M. C. Letzel, Z. Lipták, A. Pervukhin, SIRIUS: decomposing isotope patterns for metabolite identification, *Bioinf.* **25** (2009) 218–224.

[11] K. Dührkop, S. Böcker, Fragmentation trees reloaded, *J. Cheminf.* **8** (2016) #5.

[12] F. Allen, A. Pon, M. Wilson, R. Greiner, D. Wishart, CFM-ID: a web server for annotation, spectrum prediction and metabolite identification from tandem mass spectra, *Nucleic Acids Res.* **42** (2014) W94–W99.

[13] F. Allen, R. Greiner, D. Wishart, Competitive fragmentation modeling of ESI-MS/MS spectra for putative metabolite identification, *Metabolomics* **11** (2015) 98–110.

[14] F. Allen, A. Pon, R. Greiner, D. Wishart, Computational prediction of electron ionization mass spectra to assist in gc/ms compound identification, *Anal. Chem.* **88** (2016) 7689–7697.

[15] F. Allen, *Competitive Fragmentation Modeling of Mass Spectra for Metabolite Identification*, Ph.D. thesis, Univ. Alberta, 2016.

[16] J. Gasteiger, W. Hanebeck, K. P. Schultz, Prediction of mass spectra from structural information, *J. Chem. Inf. Comp. Sci.* **32** (1992) 264–271.

[17] F. Hufsky, K. Scheubert, S. Böcker, New kids on the block: novel informatics methods for natural product discovery, *Nat. Prod. Rep.* **31** (2014) 807–817.

[18] C. A. Bauer, S. Grimme, How to compute electron ionization mass spectra from first principles, *J. Phys. Chem. A* **120** (2016) 3755–3766.

[19] J. Cautereels, M. Claeys, D. Geldof, F. Blockhuys, Quantum chemical mass spectrometry: *ab initio* prediction of electron ionization mass spectra and identification of new fragmentation pathways, *J. Mass Spectrom.* **51** (2016) 602–614.

[20] C. A. Bauer, S. Grimme, First principles calculation of electron ionization mass spectra for selected organic drug molecules, *Org. Biomol. Chem.* **12** (2014) 8737–8744.

[21] W. Fontana, L. W. Buss, What would be conserved if "the tape were played twice", *P. Natl. Acad. Sci. USA* **91** (1994) 757–761.

[22] G. Berry, G. Boudol, The chemical abstract machine, *IFIP Adv. Inf. Comm. Te.* **96** (1992) 217 – 248.

[23] J. L. Andersen, C. Flamm, D. Merkle, P. F. Stadler, A software package for chemically inspired graph transformation, in: R. Echahed, M. Minas (Eds.), *Proceedings of the 9th International Conference in Graph Transformation, ICGT 2016*, Springer, Cham, 2016, pp. 73–88.

[24] H. Ehrig, K. Ehrig, U. Prange, G. Taenthzer, *Fundamentals of Algebraic Graph Transformation*, Springer, 2006.

[25] J. L. Andersen, MedØlDatschgerl (MØD), `http://mod.imada.sdu.dk`, 2016.

[26] J. L. Andersen, C. Flamm, D. Merkle, P. F. Stadler, Inferring chemical reaction patterns using rule composition in graph grammars, *J. Syst. Chem.* **4** (2013) #4.

[27] J. L. Andersen, C. Flamm, D. Merkle, P. F. Stadler, Generic strategies for chemical space exploration, *Int. J. Comput. Biol. Drug Des.* **7** (2014) 225–258.

[28] F. Allen, A. Pon, R. Greiner, D. Wishart, CFM-ID: Competitive fragmentation modeling for metabolite identification, `http://cfmid.wishartlab.com/`, 2014–2016.

[29] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.

[30] F. W. McLafferty, F. Tureček, *Interpretation von Massenspektren*, Springer, 1995.

[31] R. Gugisch, A. Kerber, A. Kohnert, R. Laue, M. Meringer, C. Rücker, A. Wassermann, MOLGEN 5.0, a molecular structure generator, in: S. C. Basak, G. Restrepo, J. L. Villaveces (Eds.), *Advances in Mathematical Chemistry and Applications 1*, Bentham, 2014, pp. 113–138.

[32] C. Flamm, D. Merkle, P. F. Stadler, U. Thorsen, Automatic inference of graph transformation rules using the cyclic nature of chemical reactions, in: R. Echahed, M. Minas (Eds.), *Proceedings of the 9th International Conference in Graph Transformation, ICGT 2016*, Springer, Cham, 2016, pp. 206–222.

[33] C. Moler, C. Van Loan, Ninteen doubious was to compute the exponential of a matrix, twenty-five years later, *SIAM Rev.* **45** (2003) 3–49.

[34] H. Rosenstock, B. Wallenstein, A. Wahrhaftig, H. Eyring, Absolute rate theory for isolated systems and the mass spectra of polyatomic molecules, *P. Natl. Acad. Sci. USA* **38** (1952) 667–678.

[35] C. E. Klots, Reformulation of the quasiequilibrium theory of ionic fragmentation, *J. Phys. Chem.* **75** (1971) 1526–1532.

[36] C. E. Klots, Quasi-equilibrium theory of ionic fragmentation: Further considerations, *Z. Naturforsch.* **27** (1972) 553–561.

[37] J. A. Herman, Y.-H. Li, A. G. Harrison, Energy dependence of the fragmentation of some isomeric $[C_6H_{12}]^{+\cdot}$ ions, *J. Mass Spectrom.* **17** (1982) 143–150.

[38] J. Marrero, R. Gani, Group-contribution based estimation of pure component properties, *Fluid Phase Equilib.* **183-184** (2001) 183–208.

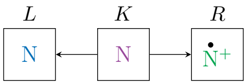
[39] E. Noor, H. S. Haraldsdóttir, R. Milo, R. M. T. Fleming, Consistent estimation of Gibbs energy using component contributions, *PLoS Comput. Biol.* **9** (2013) #e1003098.

[40] E. Noor, A. Bar-Even, A. Flamholz, E. Reznik, W. Liebermeister, R. Milo, Pathway thermodynamics highlights kinetic obstacles in central metabolism, *PLoS Comput. Biol.* **10** (2014) #e1003483.

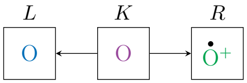
[41] G. B. Goh, N. O. Hodas, A. Vishnu, Deep learning for computational chemistry, *J. Comput. Chem.* **38** (2017) 1291–1307.

A Rules Used for Fig. 2

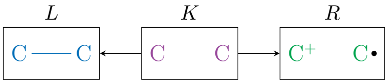
A.1 r_0, N-ionization



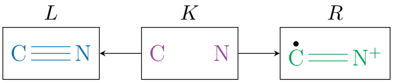
A.2 r_1, O-ionization



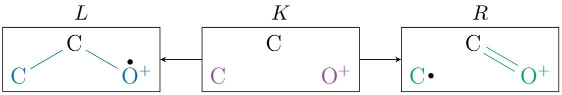
A.3 r_4, σ-ionization



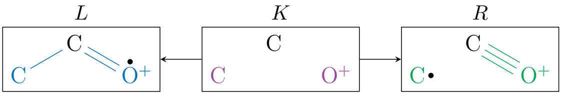
A.4 r_{10}, π-ionization



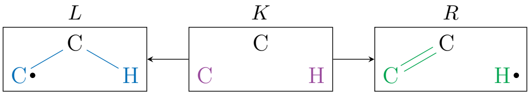
A.5 r_{14}, α-cleavage, CO, Single



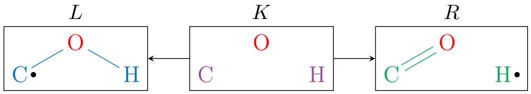
A.6 r_{28}, α-cleavage, CO, Double



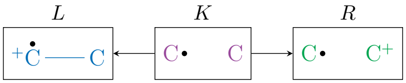
A.7 r_{34}, α-cleavage, CH



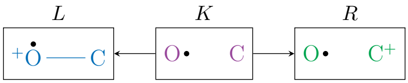
A.8 r_{35}, α-cleavage, OH



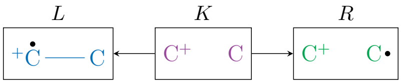
A.9 r_{38}, Inductive Cleavage, CC, 1



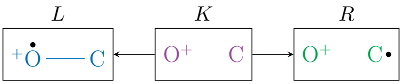
A.10 r_{40}, Inductive Cleavage, CO, 1



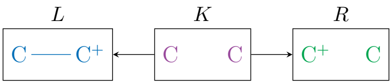
A.11 r_{43}, Inductive Cleavage, CC, 2



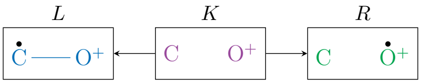
A.12 r_{45}, Inductive Cleavage, CO, 2



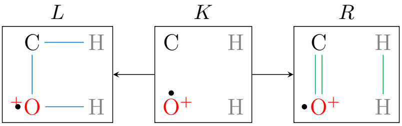
A.13 r_{52}, Inductive Cleavage, Heterolytic



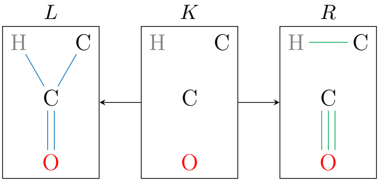
A.14 r_{59}, Inductive Cleavage, Homolytic



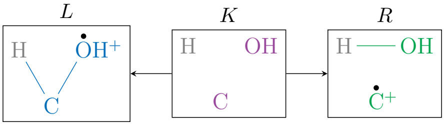
A.15 r_{71}, H_2-elimination



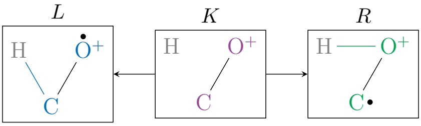
A.16 r_{75}, CO-elimination



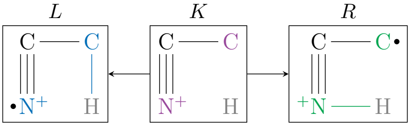
A.17 r_{77}, H_2O-elimination



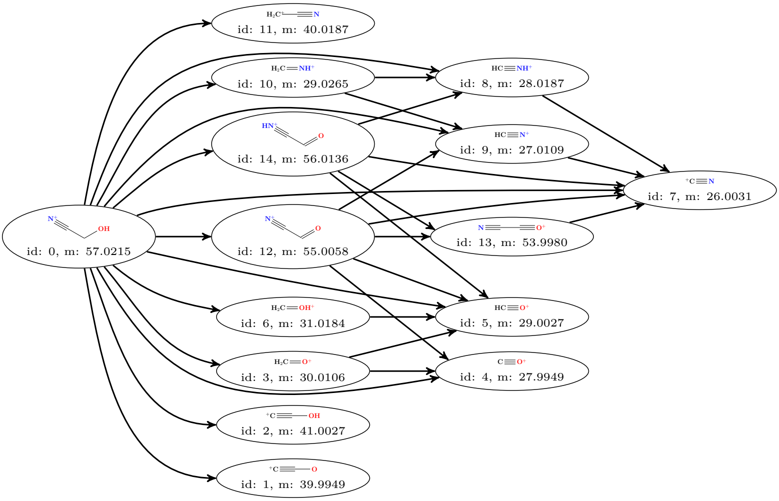
A.18 r_{88}, 1,2 H-shift



A.19 r_{96}, H-rearrangement



B Fragmentation Graph by CFM-ID



This fragmentation graph for glycolonitrile was produced by the module `fraggraph-gen` of CFM-ID [28] with the arguments SMILES: N#CCO, depth 4, ionization mode: positive EI, and Graph: fullgraph. It illustrates the propagation of errors from the fragmentation process to the predicted spectrum: Of the two most abundant peaks (m/z 28 and m/z 18) in the measured spectrum of glycolonitrile, Figure 3, only m/z 28 (id 8) is present in the fragmentation graph produced by `fraggraph-gen`. If the mass spectrum is now simulated based on this fragmentation graph using the module `cfm-predict` in CFM-ID, only 6 fragments survive and accumulate intensity. The three most abundant peaks in the predicted spectrum are located at m/z 57, 40, and 31 with relative intensities 0.59, 0.13, and 0.09. The fragment m/z 28 does not accumulate any intensity. Hence the predicted spectrum completely lacks the two peaks that are most abundant in the experiment.

C Example Strategies for Fragmentation Graphs

Continuing from Sec. 4, we give some more examples of strategies for creation of fragmentation graphs.

The following is an example strategy for the case where molecules including a ring structure have one set of valid fragmentation rules and molecules without any rings have another. This is modeled by a parallel strategy in each iteration:

```
$\texttt{parallel}[R_{\text{ionization}}] \rightarrow \texttt{filter}[P_{\text{charge}}] \rightarrow \texttt{repeat}[4,$
    $\texttt{parallel}[$
        $\texttt{filter}[P_{\text{ring}}] \rightarrow \texttt{parallel}[R_{\text{fragmentation,ring}}] \rightarrow \texttt{filter}[P_{\text{charge}}],$
        $\texttt{filter}[P_{\text{tree}}] \rightarrow \texttt{parallel}[R_{\text{fragmentation,tree}}] \rightarrow \texttt{filter}[P_{\text{charge}}],$
    $]$
$]$
```

where P_{ring} and P_{tree} are predicates that decide if a graph has respectively at least one ring and no rings.

A variation of this scenario is when all compounds must be fragmented with one set of rules, but compounds with ring structures are subjected to an additional set of fragmentation rules:

```
$\texttt{parallel}[R_{\text{ionization}}] \rightarrow \texttt{filter}[P_{\text{charge}}] \rightarrow \texttt{repeat}[4,$
    $\texttt{parallel}[$
        $\texttt{filter}[P_{\text{ring}}] \rightarrow \texttt{parallel}[R_{\text{fragmentation,ring}}] \rightarrow \texttt{filter}[P_{\text{charge}}],$
        $\texttt{parallel}[R_{\text{fragmentation,tree}}] \rightarrow \texttt{filter}[P_{\text{charge}}],$
    $]$
$]$
```

In the final example we introduce a general hooking mechanism for constraining fragmentation. It may, for instance, be used when we have additional knowledge from external sources or programs that some types of fragmentation do not occur in practice, but in a way that cannot be translated into extensions of the graph transformation rules.

```
$\texttt{derivationPredicate}[P_{\text{checkIonization}},$
    $\texttt{parallel}[R_{\text{ionization}}]$
$] \rightarrow \texttt{filter}[P_{\text{charge}}]$
$\rightarrow \texttt{derivationPredicate}[P_{\text{checkFragmentation}},$
    $\texttt{repeat}[4,$
        $\texttt{parallel}[$
            $\texttt{filter}[P_{\text{ring}}] \rightarrow \texttt{parallel}[R_{\text{fragmentation,ring}}] \rightarrow \texttt{filter}[P_{\text{charge}}],$
                              $\texttt{parallel}[R_{\text{fragmentation,tree}}] \rightarrow \texttt{filter}[P_{\text{charge}}],$
        $]$
    $]$
$]$
```

Using MØD [23, 25] this can be written in the following way:

```
targetCompounds = [smiles("N#CCO")]

def checkIonization(d):
   # do checks
   # if checks fail: return False
   # else
   return True

def checkFragmentation(d):
   # do checks
   # if checks fail: return False
   # else
   return True

def hasCharge(g, gs, first):
   return sum(v.charge for v in g.vertices) != 0

def hasRing(g, gs, first):
   return g.numEdges >= g.numVertices

strat = rightPredicate[checkIonization](
   ionizationRules
   >> filterSubset(hasCharge)
   >> rightPredicate[checkFragmentation](
      repeat[4](  [
            filterSubset(hasRing) >> ringFragmentationRules >> filterSubset(hasCharge),
            treeFragmentationRules >> filterSubset(hasCharge)
      ]  )
)  )
dg = dgRuleComp(inputGraphs, addSubset(targetCompounds) >> strat)
dg.calc()
dg.print()
```