Reaction Rebalancing: A Novel Approach to Curating Reaction Databases

Tieu-Long Phan[1,2*†], Klaus Weinbauer[1,3†], Thomas Gärtner[3], Daniel Merkle[4,2], Jakob L. Andersen[2], Rolf Fagerberg[2], Peter F. Stadler[1,5,6,7,8,9]

[1]Bioinformatics Group, Department of Computer Science & Interdisciplinary Center for Bioinformatics & School for Embedded and Composite Artificial Intelligence (SECAI), Leipzig University, Härtelstraße 16–18, D-04107 Leipzig, Germany.
[2]Department of Mathematics and Computer Science, University of Southern Denmark, DK-5230 Odense M, Denmark .
[3]Machine Learning Research Unit, TU Wien Informatics, Erzherzog-Johann-Platz 1 (FB02), A-1040 Wien, Austria.
[4]Faculty of Technology, Bielefeld University, Postfach 10 01 31, D-33501 Bielefeld, Germany.
[5]Max Planck Institute for Mathematics in the Sciences, Inselstraße 22, D-04103 Leipzig, Germany.
[6]Department of Theoretical Chemistry, University of Vienna, Währingerstraße 17, A-1090 Wien, Austria.
[7]Facultad de Ciencias, Universidad National de Colombia, Bogotá, Colombia.
[8]Center for non-coding RNA in Technology and Health, University of Copenhagen, Ridebanevej 9, DK-1870 Frederiksberg, Denmark.
[9]Santa Fe Institute, 1399 Hyde Park Rd., Santa Fe, NM 87501, USA.

*Corresponding author(s). E-mail(s): long.tieu_phan@uni-leipzig.de;
Contributing authors: klaus@bioinf.uni-leipzig.de; thomas.gaertner@tuwien.ac.at; daniel.merkle@uni-bielefeld.de; jlandersen@imada.sdu.dk; rolf@imada.sdu.dk; studla@bioinf.uni-leipzig.de;

[†]These authors contributed equally to this work.

Abstract

Purpose: Reaction databases are a key resource for a wide variety of applications in computational chemistry and biochemistry, including Computer-aided Synthesis Planning (CASP) and the large-scale analysis of metabolic networks. The full potential of these resources can only be realized if datasets are accurate and complete. Missing co-reactants and co-products, i.e., unbalanced reactions, however, are the rule rather than the exception. The curation and correction of such incomplete entries is thus an urgent need.

Methods: The `SynRBL` framework addresses this issue with a dual-strategy: a rule-based method for non-carbon compounds, using atomic symbols and counts for prediction, alongside a Maximum Common Subgraph (MCS)-based technique for carbon compounds, aimed at aligning reactants and products to infer missing entities.

Results: The rule-based method exceeded 99% accuracy, while MCS-based accuracy varied from 81.19% to 99.33%, depending on reaction properties. Furthermore, an applicability domain and a machine learning scoring function were devised to quantify prediction confidence. The overall efficacy of this framework was delineated through its success rate and accuracy metrics, which spanned from 89.83% to 99.75% and 90.85% to 99.05%, respectively.

Conclusion: The `SynRBL` framework offers a novel solution for recalibrating chemical reactions, significantly enhancing reaction completeness. With rigorous validation, it achieved groundbreaking accuracy in reaction rebalancing. This sets the stage for future improvement in particular of atom-atom mapping techniques as well as of downstream tasks such as automated synthesis planning.

Keywords: reaction databases, unbalanced reactions, data curation, `SynRBL`, rules, maximum-common-subgraph

1 Introduction

Large-scale reaction databases such as the United States Patent and Trademark Office (`USPTO`) database [1] and the commercial database `Reaxys` [2] cataloge millions of chemical reactions and serve to enable data-driven approaches in chemistry. `Reaxys`, hosting over 55 million manually curated reactions, has become a cornerstone for deploying deep-learning neural networks in retrosynthesis [3, 4, 5, 6, 7], robotic chemistry [8], and the determination of optimal reaction conditions [9].

`USPTO` is the largest public collection of chemical reactions, comprising more than 3 million entries mined from approximately 9 million US patents covering 1976 to 2016. Its impact on cheminformatics and synthetic chemistry is significant, and as a public resource, it has particular impact in methods development. It plays a pivotal role in the advancement of reaction database analysis [10], forward [11, 12, 13] and backward [14] synthesis prediction, and yield prediction [15, 16]. The database has been instrumental also in reaction classification [17, 18], atom-to-atom mapping [19, 20], and synthesis rule clustering [21].

Despite the rapid advancements of databases, data quality remains a significant issue in particular for machine learning applications in chemistry [22]. A particularly serious problem is that omission of co-reactants or co-products. For example, less than 12% of the single step reactions in Reaxys analyzed to study the exploration history of chemical space [23] were balanced. This problem has multiple roots, including historical and procedural practices. These deficiencies are attributed to the limitations of text mining, which struggles with the variability of publication formats [24], and to errors introduced during manual data curation [25].

Many data-driven applications therefore attempt to ignore the fact that many or most reactions are unbalanced and operate directly on such imperfect reaction data. This is in particular the case of atom-atom mapping methods. RXNMapper [20] and GraphormerMapper [26] apply machine learning for reaction mapping and atom embedding improvements, respectively, without directly addressing reaction imbalances. Jaworski's rule-based atom-atom-mapper [19], on the other hand, uses graph-theoretic considerations that introduce small molecules to achieve stoichiometric balance before atom correspondences are inferred. GraphormerMapper was reported to show enhanced performance on the Golden dataset of manually mapped and curated reactions [27]. Its efficacy on unbalanced reactions remains undocumented.

Several tools dedicated to balacing reactions have become available. CGRTools offers a rule-based method for rebalancing reactions by adding small molecules, which however has limited success in achieving perfect balance [28]. A hybrid workflow [29] combines ChemBalancer's heuristic methods and ChemMLM's machine learning to enhance molecule prediction. While ChemBalancer focuses on reaction completion, lacking precise accuracy metrics, ChemMLM shows promise with small molecules but struggles with complex structures [29].

The SynRBL framework for rebalancing reactions, which we introduce here, combines two methods: a rule-based approach for missing non-carbon compounds, i.e. compounds without carbon atoms like H_2O or HCl, and a graph-theoretic approach for missing carbon structures. The rule-based method uses atomic symbols and counts to determine if reactions are balanced, decomposing molecules into ions to minimize redundancy and employing a search strategy that leverages a rule library to identify missing molecules.

For carbon compounds, we consider a maximum common subgraph (MCS) problem. This family of combinatorial optimization problems plays an important role in structural comparisons in chemistry and biology [30]. It underlies similarity searches vital to the preliminary phases of drug discovery, offering metrics for molecular structure similarity based on MCS dimensions, in alignment with the principle of similar properties [31, 32]. Beyond similarity assessment, MCS analysis is integral to clustering processes [33, 34, 35], the identification of matched molecular pairs [36], reaction mapping [37, 38], and the alignment of molecules [39]. MCS problems come in two flavors, both of which are NP-hard [40]. These two flavors are the maximum common induced subgraph (MCIS), which focuses on atom count, and the maximum common edge subgraph (MCES), which focuses on edge count. They give notable differences in the analyses of dissimilar molecules [41]. Our MCS-based approach targets carbon compound gaps and reactions beyond the rule-based method's scope by aligning reactants and products to pinpoint and merge non-aligned segments, generating missing compounds. An iterative technique proceeding by overlapping molecules one at a time and isolating non-overlapping regions for efficient alignment in subsequent rounds is introduced to reduce computational costs.

2 Method

2.1 Notation and Preliminaries

Every chemical reaction r can be written in the form

$$\sum_i s^-_{ir} X_i^{(q^-_{ir})} \to \sum_j s^+_{jr} X_j^{(q^+_{jr})} \tag{1}$$

where $s^-_{ir} \geq 0$ and $s^+_{jr} \geq 0$ are the stochiometric coefficients of compounds X_i and X_j appearing as a reactant and as product, respectively. The superscripts (q^-_{ir}) and (q^+_{jr}) indicate the charge of the compounds X_i and X_j among the reactants and products, respectively. A molecule does not appear as a reactant or product if its stoichiometric coefficient vanishes, i.e., if $s^-_{ir} = 0$ and $s^+_{jr} = 0$, respectively. Since we consider only a single fixed reaction in the following, we drop the index r from here on.

Every compound X_i has a well-defined composition expressed by its sum formula. We write n_{ai} for the number of atoms of type a in compound i. The equilibrium of chemical reactions, grounded in the Law of Conservation of Mass by Antoine Lavoisier [42], stipulates that all reactions r are *balanced* in the sense that the total number n^-_{ar} of atoms of type a in the reactants equals the total number n^+_{ar} of atoms of type a in the products, i.e.,

$$n^-_a := \sum_i n_{ai} s^-_i = \sum_i n_{ai} s^+_i =: n^+_a \tag{2}$$

Similarly, the Law of Conservation of Charge ensures the constancy of total charge, crucial in redox and ionic reactions, i.e., it ensures that for every reaction

$$q^- := \sum_i s^-_i q^-_i = \sum_i s^+_i q^+_i =: q^+ \tag{3}$$

In organic chemistry, carbon balancing (expressed as $n^-_C = n^+_C$), is essential for tracking carbon atoms in bond formations or cleavages, highlighting the significance of carbon atom accounting [43]. Balancing carbons is in practice more challenging because the imbalance is usually much larger compared to the atoms found in functional groups because larger organic molecules are not represented in the reaction data.

The task of reaction balancing can be expressed as follows. If a reaction is *unbalanced*, i.e., if $n^-_a \neq n^+_a$ for one or more atom types a, find a set of reactants $\{X_k^{(q^-_k)}\}$ and a set of products $\{X_l^{(q^+_l)}\}$ with non-zero stoichiometric coefficients t^-_k and t^+_l such that

$$n_a^- + \sum_k n_{ak} t_k^- = \sum_l n_{al} t_l^+ + n_a^+ \tag{4}$$

holds for all atom types a and, likewise, the charges satisfy

$$q^- + \sum_k t_k^- q_k^- = \sum_l t_l^+ q_l^+ + q^+ \tag{5}$$

The practical complication is that (i) the set of possible compounds that may appear as additional reactants or products is too large for brute force enumeration, and (ii) even if this were possible, not all choices that formally might solve the problem are chemically plausible. To simplify the notation further, we can treat the charge as an additional formal "atom type" that may take on both positive and negative integer values, corresponding to positive and negative charges, respectively. This amounts to considering free electrons e^- as a special compound. Moreover, we write n_q^- and n_q^+ instead of q^- and q^+ for the net charge in the following. Note that by convention a free electron e^- corresponds to a charge of -1. In the remainder of this section, we describe two alternative strategies for rebalancing chemical reactions.

2.2 Rule-based Method

2.2.1 Representation of Molecules and Reactions

It is common well-known issue that entries in reaction databases often omit one ore more simple compounds such as H_2O, NH_3, and HCl.

To rebalance such incompete reaction data, we developed a specialized rule library to systematically incorporate these missing elements utilizing the cheminformatics library `RDKit` 2023.9.4 [44]. To facilitate computations, we represent the sum formula of molecules as a dictionary.

$$\mathcal{D} := \{C_1 : n_1, C_2 : n_2, \dots, C_\ell : n_\ell, Q : n_Q\}$$

Here, each C_a, $1 \le a \le l$, is an atomic symbol, i.e., H, O, or N, and $n_a \in \mathbb{N}$ is the number of atoms of type C_a in the compound under consideration. We use the special symbol Q to denote charge associated with the molecule. Recall that $n_Q \in \mathbb{Z}$ can be positive, negative, or zero.

The rule-based strategy is applied only to reactions that are carbon-balanced. The reason is that in organic reactions, the structure of the carbon backbone plays a key role, and thus, sum formulas are much less likely to be sufficient to completely describe the missing molecules. We also optimized our approach by considering the standard representation of ions in chemical equations, such as OH^- and H^+, instead of NaOH or HCl. To achieve this, we restructured our rule library to focus on elementary ions, enabling us to interpret compounds such as HCl in terms of their constituent ions, H^+ and Cl^-. This refinement led to a more efficient and compact rule library, as depicted in Table S3.

We denote by $\mathcal{D}^-$ and $\mathcal{D}^+$ the composition dictionaries of the sum of the molecular formulae of reactants and products, respectively. That is, $\mathcal{D}^-$ has entries of the form

$C_a : n_a^-$, and $\mathcal{D}^+$ has entries $C_a : n_a^+$. The discrepancy between $\mathcal{D}^-$ and $\mathcal{D}^+$ is conveniently represented by two dictionaries Δ^+ with entries $C_a : n_a^+ - n_a^-$ provided $n_a^+ > n_a^-$, and Δ^- with entries $C_a : n_a^- - n_a^+$ provided $n_a^+ < n_a^-$. Thus Δ^+ accounts for the atoms only present in the products and Δ^- accounts for the atoms only present in the reactants.

Based on the difference dictionaries $\Delta^\pm$ we distinguish four cases:

- *balanced* if $\Delta^+ = \Delta^- = \emptyset$,
- *reactant-dominated* if $\Delta^- \neq \emptyset$ and $\Delta^+ = \emptyset$,
- *product-dominated* if $\Delta^+ \neq \emptyset$ and $\Delta^- = \emptyset$,
- *both-sides* if both $\Delta^- \neq \emptyset$ and $\Delta^+ \neq \emptyset$.

If only one of Δ^- and Δ^+ has a non-charge entry, then the charge difference is accounted for in the same dictionary, while the other one is left empty. This is always possible since charges may be positive or negative. Instances of the both-sides case, i.e., instances with missing atoms in both reactants and products are not considered further here. They require a more sophisticated approach and are relegated to the MCS-based method in our current implementation.

Reactant-dominated and product-dominated cases are handled in the same manner. In the following, we denote by Δ the single non-empty difference dictionary.

For example, the database entry

$$\mathrm{CH_3COOH} + \mathrm{C_2H_5OH} \longrightarrow \mathrm{CH_3COOC_2H_5}$$

yields the dictionaries $\mathcal{D}^- = \{\mathrm{C}:4, \mathrm{H}:10, \mathrm{O}:3\}$ and $\mathcal{D}^+ = \{\mathrm{C}:4, \mathrm{H}:8, \mathrm{O}:2\}$ for the reactants and product, respectively, and thus $\Delta^- = \{\mathrm{O}:1, \mathrm{H}:2\}$.

2.2.2 Molecular Imputation

For ease of presentation we assume $\Delta = \Delta^-$, i.e., atoms are missing on the product side only. Otherwise, the role of reactants and products is interchanged.

We consider a set $\mathcal{R}$ of rules that explain (part of) the dictionary Δ in terms of molecules X_k that are added to the product side. Our goal is to find a sequence of rule applications which stepwise reduce the difference dictionary Δ and collect a multiset S of molecules. Each $r \in \mathcal{R}$ is of the form $\hat{r} \rightsquigarrow X_r$, where $\hat{r}$ is a dictionary and X_r is a corresponding molecule. The application of a rule changes Δ accordingly. Since our rules make use of simple ions, we allow arbitrary changes of charges. The rule

$$\{\mathrm{O}:1, \mathrm{H}:1, \mathrm{Q}:-1\} \rightsquigarrow \mathrm{OH}^-$$

applies to dictionary $\Delta = \{\mathrm{O}:1, \mathrm{H}:2\}$ by adding OH^- to the products and updating the dictionary to $\Delta = \{\mathrm{H}:1, \mathrm{Q}:1\}$. The resulting reaction is still unbalanced and reactant-dominated, hence another rule may apply.

If we reach $\Delta = \emptyset$, then adding S to the products balances the reaction. In practice, this can be achieved by the basic DFS search [45] outlined in Alg. 1. A call to DFS(Δ, $\mathcal{R}$, $\emptyset$) either returns all (multi)sets of compounds S that balances the reaction and leaves an empty dictionary Δ, or it terminates without output. By $\Delta \ominus \hat{r}$ we denote the dictionary Δ after being modified by the application of a rule r.

Algorithm 1 DFS-like rule application

```
function DFS($\Delta$, $\mathcal{R}$, $S$)
    if $\Delta = \emptyset$ then
        Yield $S$
    else
        for each rule $(\hat{r} \rightsquigarrow X_r) \in \mathcal{R}$ applicable to $\Delta$ do
            $\Delta' \leftarrow \Delta \ominus \hat{r}$        $S' \leftarrow S \cup \{X_r\}$
            DFS($\Delta'$, $\mathcal{R}$, $S'$)
        end for
    end if
end function
```

The DFS algorithm yields all balancing solutions. These are passed on to the post-processing step (2.2.3). The list $\mathcal{R}$ of rules is applied in a fixed order that ensures that pattern size, defined as the number atoms in $\hat{r}$, is non-increasing. Thus, the search can be restricted to check only patterns with a valid length. One could use the fact that the dictionary obtained by the successful application of several rules is independent of the order in which these rules a applied. Keep track of the rule r that was applied before $\text{DFS}(\Delta, \mathcal{R}, S)$ was called it therefore suffices to disregard in the next recursion step all rules that appear before r in $\mathcal{R}$. Moreover, one could abandon a recursion step if its path length exceeds the best previously found solution. The latter modification however limits the scope of post-processing rules intended to remove chemically implausible solutions. Since simple DFS is already comparably fast and the search tree is usually quite shallow, such optimization are currently not implemented.

Continuing the example, after the first match, we may apply the rule $\{\text{H} : 1, \text{Q} : 1\} \rightsquigarrow \text{H}^+$, which leaves the dictionary Δ empty. The DFS function first gives $S = \{\text{OH}^-, \text{H}^+\}$ and we arrive a the (chemically correct) balanced reaction

$$CH_3COOH + C_2H_5OH \longrightarrow CH_3COOC_2H_5 + OH^- + H^+.$$

In general, there will be multiple solutions. Thus, continuing the DFS after it yields the first result turns it into an exhaustive search. The advantage of listing all solutions is that they can be evaluated, and an optimal solution can be identified. Here, we use the minimal number of rules as an optimization criterion. This favors matches of large partial dictionaries. When multiple solutions exhibit an equivalent minimal count of rules ascertained through the DFS algorithm, precedence is accorded to the solution that encompasses an ion in the set S.

2.2.3 Post-processing

In some cases, the balancing of a reaction using DFS(Δ, $\mathcal{R}$, $\emptyset$) yields a formally correct solution that is chemically implausible. More precisely, S may contain one or more molecules that are at least unlikely to be the true reactants or products. In some cases, it is possible to find a more plausible rebalancing. Oxygen and halogens are typically formed via potent oxidizing agents. Hydrogen, on the other hand, is usually produced in reactions with alkali metals (e.g., lithium, sodium, potassium) or hydride compounds. Whether this is the case can be checked after DFS(Δ, $\mathcal{R}$, $\emptyset$) has successfully balanced the reaction. Currently, SynRBL considers only three post-processing rules:

(i) If a free halogen appears as a product, we assume that the solution is invalid and reject the completion.

(ii) If oxygen O appears as a product, we add H_2 as a missing reactant and replace O by H_2O on the product side.

(iii) If hydrogen H_2 appears on the product side and there is neither an alkali metal nor a hydride among the reactant, we add O to the reactants and replace H_2 by H_2O on the product side.

The software is designed in a manner that makes it straightforward to extend this rule set.

2.2.4 Redox Reaction Refinement

Consider the reduction reaction involving the transformation of acetic acid into ethanol: $CH_3COOH \longrightarrow C_2H_5OH$. The rule-based methodology aptly addressed this reaction by introducing two moles of hydrogen H_2 to the reactant side and one mole of water (H_2O) to the product side, thereby yielding the stoichiometric equation:

$$CH_3COOH + 2\,H_2 \longrightarrow C_2H_5OH + H_2O$$

It is essential to acknowledge that the depicted reaction is not viable due to the insufficient reactivity of molecular hydrogen (H_2) for the reduction of acetic acid. Typically, this reaction necessitates a suitable reducing agent, such as lithium aluminum hydride ($LiAlH_4$). However, identifying and substituting the appropriate reducing agents can be problematic. Some chemists use a convention to simplify chemical notations where the reducing agent is represented as [H] without specifying the exact compound. Following this convention, we have updated the notation from molecular hydrogen (H_2) to two single hydrogen atoms (H). This new representation indicates the presence of a reducing agent distinct from elemental hydrogen. Likewise, the depiction of molecular oxygen as O_2 has been revised to two single oxygen atoms (O), symbolizing its role as an oxidizing agent.

2.3 MCS-based method

2.3.1 Determination of Missing Carbon Compounds

Carbon-unbalanced reactions cannot be meaningfully handled at the level of sum formulas. Instead, it is necessary to make use of the structures of reactant and product molecules. To this end, we represent both the reactants and the products of a reaction as graphs whose connected components are the molecules. In these graphs, vertices are labeled by atom types and edges correspond to chemical bonds, annotated by their bond type. Since reactions with carbon atoms missing on the reactant side are treated in the same way as reactions with missing carbon on the product side, we fix the notation as follows:

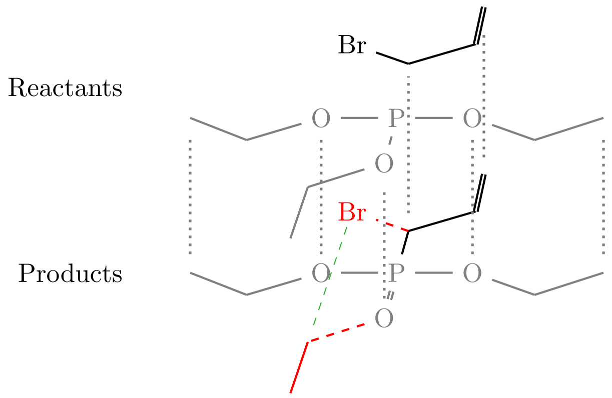

Fig. 1: In this example two fragments (shown in red) remain unmatched: Br with a single bond as cut, and an ethyl group also with a single bond. The cut edges of the fragments are show as dashed red lines. A merge rule insert a single bond (dashed green) connecting the end-points of the cut edges.

Let X and Y be the graphs with the larger and smaller number of carbons, respectively. Moreover, we write $\mathcal{X} = \{X_1, X_2, \dots, X_k\}$ for the set of connected components of X. Assuming that all missing carbons belong to one connected compound Y_* missing on the Y-side of the reaction, we can conclude that Y_* is in essence a part of some X_i. In order to identify this part, we compute, for each $X_i \in \mathcal{X}$, a maximum *connected* common subgraph $M_i = \mathrm{MCS}(X_i, Y)$. There are several choices for the exact definition of the function $\mathrm{MCS}(\,.\,)$, which we will discuss in more detail below. For the moment we only require that the subgraph M_i is connected and that $\mathrm{MCS}(\,.\,)$ defines an injective map of the vertex set $V(M_i)$ into $V(X_i)$ and $V(Y)$ where each vertex in $V(Y)$ is only mapped once. We can therefore identify the vertices of M_i with a subset of the vertices of X_i and, by a slight abuse of notation, simply write $V(M_i) \subseteq V(X_i)$. This, in turn, specifies a (bipartite) matching between vertices of X_i and Y that correspond to the same vertex of M_i. In chemical terms, this matching is a partial atom-atom map between X_i any Y and thus also between X and Y. To characterize the part of X_i that does not match Y in more detail, we consider the subgraph $A_i \coloneqq X_i[V(X_i) \setminus V(M_i)]$ of X_i induced by the unmatched vertices. Moreover, let B_i be the edge cut between $V(A_i)$ and $V(M_i)$ in X_i. In chemical terms, B_i denotes the bonds that separate M_i and A_i and thus were broken (or formed) by the reaction. A vertex in A_i is said to be a *boundary vertex* if it is incident to a cut edge $e \in B_i$.

Denote by $\mathcal{A} \coloneqq \{(A_i, B_i) | X_i \in \mathcal{X}\}$ the set of auxiliary graphs together with their separating edge cuts. We shall refer to these as *fragments*. By construction, $\mathcal{A}$ contains the relevant information on the mission compounds because the union $\bigcup_i V(A_i)$ is the set of missing atoms, and the B_i are bonds on X_i that are broken in order to obtain Y. The task at hand, therefore, is to "recombine" the (A_i, B_i) in a way that recovers the missing compound(s) Y_*. To this end, we again pursue a rule-based approach. We consider two types of rules:

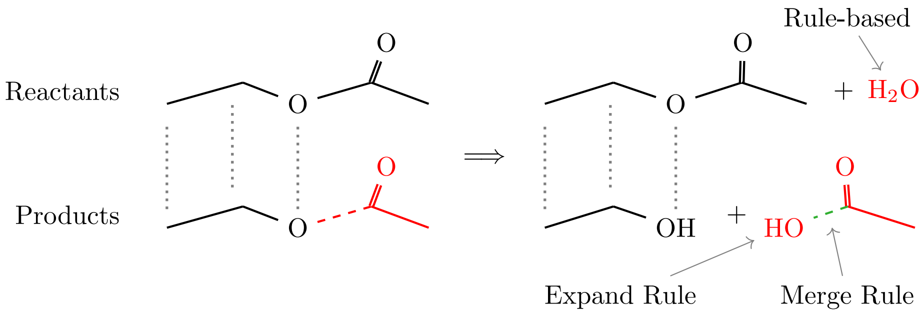

Fig. 2: Graph alignment and imputation of missing parts (red). The absence of the second reactant is solved by applying an *expand rule* before merging the fragments with an appropriate *merge rule*.

Merge rules encode conditions for the insertion of edges between two boundary vertices, $u \in V(A_i)$ and $v \in V(A_j)$ located in distinct fragments, see Fig. 1. These rules depend on the specific boundary configuration, i.e., the chemical context of the two boundary atoms u and v. The application of a merge rule not only inserts a bond (labeled edge) between u and v, but also removes the respective cut edges incident to u and v from B_i and B_j, respectively. Thus only one merge rule is applied for each boundary. The boundaries are then considered resolved in the chemical domain. Moreover, open boundaries on the same compound are never merged with each other. Hence, this step always needs at least two compounds. If only one is available, *expand rules* are applied first to add the missing second fragment. A collection of merge rules is provided as configuration file and can easily be extended or modified in `SynBRL`. Table S1 in the supplementary lists the currently implemented merge rules. The alignment and imputation on a simple example are depicted in Fig. 1.

Expand rules are used to add nodes to the molecular graph based on the boundary configuration of unmatched fragments. More precisely, they can add fragments with boundaries to $\mathcal{A}$ depending on what is needed for unresolvable cut edges. This is in particular the case if $\mathcal{A}$ comprises only a single fragment (A, B). The idea of the expand rules is to add additional atoms such that cut edges that do not have a counterpart in another fragment are "saturated". Technically, however, an *expand rule* only adds the required atom, and the actual bond is then formed by a *merge rule*. Expand rules are also specified in a configuration file. Table S2 in the supplementary lists the currently implemented merge rules.

Each application of a merge step reduces the number of cut edges in the fragment set $\mathcal{A}$. Repeated rule application either terminates prematurely with no further applicable rule, or it succeeds replacing all cut-edges, thus resulting in a graph Z without remaining boundary vertices. By construction, the reaction $X \to Y \cup Z$ is now carbon balanced. It is not balanced in general. Note that the expand steps have added additional non-carbon atoms.

In practice, most carbon unbalanced reactions are missing a structure at the product side of the reaction. Hence, the methodology focuses on reactant-dominant

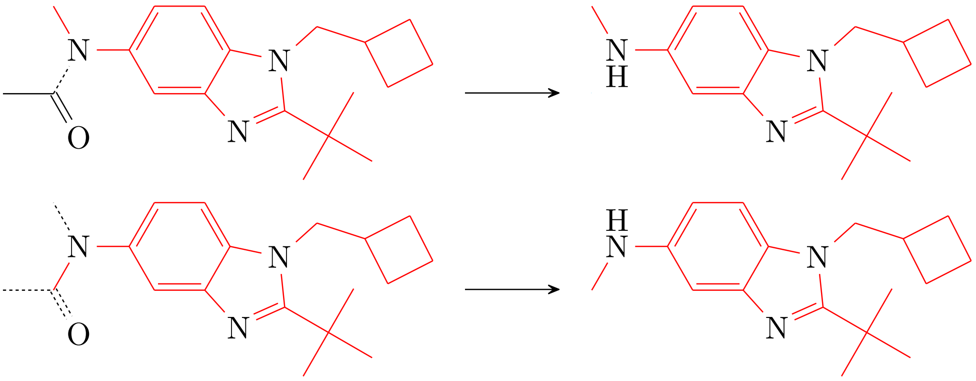

Fig. 3: Example of the ambiguity of the MCS. The product has two distinct isomorphisms in the reactant. The first example has one resulting fragment, and the second has three fragments. Dotted lines indicate the broken bonds.

reactions. In principle, it can be applied to product-dominant reactions as well. However, imputing a missing reactant is more challenging than finding a missing product. A single reaction equation can often contain multiple reaction steps, leading to multiple equally correct intermediate compounds that could be added to the reactants to form a balanced reaction. Since these cases are of minor practical relevance, we have no attempted to formulate specific rules for product-dominant reactions.

Fig. 2 shows a simple de-esterification as an example. Here, only one missing fragment is detected. Because the carbon-oxygen bond is part of an ester group, an expand rule adds the missing oxygen atom to the reaction. In the second step, a merge rule connects this oxygen with a single bond to the open boundary on the identified fragment, creating the missing acetic acid. The resulting reactions is carbon balanced but unbalanced overall. The rule-based method described in Section 2.2 is now applicable to add the missing water molecule to the reactants.

2.3.2 Computing Maximum Common Molecular Subgraphs

Maximum common subgraph (MCS)problems come in different variants. Both the maximum common induced subgraph (MCIS) problem and the maximum common edge subgraph (MCES) problem, as well as their restrictions to connected common subgraphs, are NP-hard [40]. Nevertheless they can be solved efficiently for small pairs, and thus also for molecules. However, none of the variants of combinatorial optimization problem is guaranteed to identify the “chemically correct” common subgraph, i.e., the one that correctly identifies all bonds that change during a chemical reaction.

While the size of an MCS is uniquely defined, neither the common subgraph nor its embedding is unique in general. In the example in Fig. 3 the subgraph isomorphism for the red subgraph is not unique. This is a well-known issue for the construction of atom-atom-mapping tools. These ambiguities are not easily resolved because the combinatorial MCS problems operate on graphs rather than a more detailed model of the molecules that encompasses also e.g. hybridization or partial charges.

In order to improve over the application of any one particular problem variant or algorithm, `SynRBL` resorts to the heuristics implemented in `RDKit` [44] and computes several alternative variants: MCISis addressed using the Fragment Matching and Compound Similarity (FMCS) [46], while the Rascal algorithm [47], as implemented in the `RDKit` library, is used to solve the MCES problem. Moreover, an ensemble method that amalgamates outcomes from five distinct configurations, detailed in Table 1 is used. Each of these specifies additional constraints on the matches allowed in the corresponding MCIS or MCES variant. Both the `RingMatchesRingOnly` and the `CompleteRingsOnly` ensure that atoms in rings match atoms in rings only. In graph-theoretical terms this corresponds to singling out the vertices in non-trivial 2-connected components. With the latter option, rings must be matched completely. In addition, bond order (treated as edge label) can be used as a constraint to prohibit the matching on single and double bonds.

Table 1: MCS Configuration

Configuration	1	2	3	4	5
`RingMatchesRingOnly`	True	True	False	False	-
`CompleteRingsOnly`	True	True	False	False	-
`Ignore Bond Order`	True	False	True	False	-
Algorithm	FMCS	FMCS	FMCS	FMCS	RASCAL

In order to deal with alternative embeddings of the MCS, we enumerate all maximal solutions of MCS(X_i, Y) and identify the solutions that minimize the number of fragments resulting from the removal of the common subgraph. In the example in Fig. 3, one isomorphism corresponds to the disruption of the amide bond CO$-$N, thereby producing one additional fragment. The alternative embedding of the same common subgraph implies breaking bonds containing the amine bond CH_3-N, resulting in three additional fragments. Hence, we choose the former embedding.

In order to keep the computational costs low, we do not compute MCS(X, Y) directly, but instead use an iterative approach that successively aligns the components $X_i \in \mathcal{X}$ and removes the matched vertices from Y. More precisely, for each $X_i \in \mathcal{X}$ we compute MCS$(X_i, Y^{(i-1)}))$ and construct $Y^{(i)}$ by removing all matched vertices from $Y^{(i-1)}$. To do this efficiently, we sort $\mathcal{X}$ in order of decreasing number of vertices in the connected components. As part of each evaluation of MCS$(X_i, Y^{(i-1)}))$ we also keep track of the cut edges between the matched and unmatched vertices, i.e., the broken bonds, which in particular allows us to compute the (A_i, B_i) from the iterative MCS approach.

2.4 Interaction of the two Methods

The rule-based method offers efficient solutions for non-carbon compounds, whereas the MCS-based approach focuses on subgraphs to find missing carbon structures. Identifying the optimal common subgraph is computationally intensive, making the MCS-based method less suitable for non-carbon compounds. Consequently, applying the two methods complementarily, each to their respective optimal scenarios, enhances

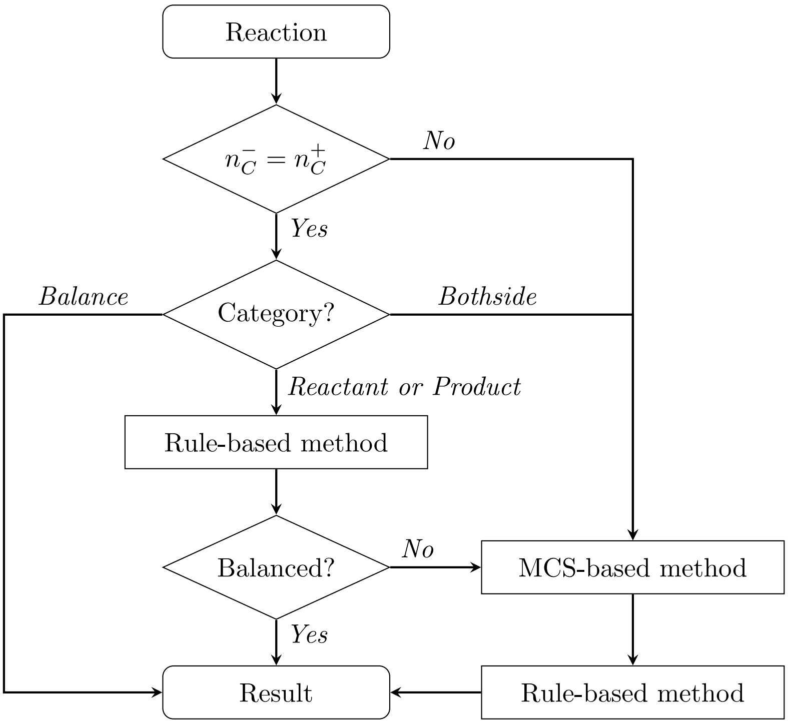

Fig. 4: Simplified overview of the functional process in `SynRBL`. The rule-based method is applied if the reaction is carbon-balanced but otherwise unbalanced in either the reactant or the product side. The MCS-based method is used if both sides are unbalanced, the rule-based method fails, or the reaction has a carbon imbalance in the first place. The output is either the balanced reaction if the method is successful or the unmodified input in case `SynRBL` can not find a solution.

overall efficiency: the rule-based approach for non-carbon compounds and the MCS-based method for situations where subgraph analysis is advantageous. The overall framework is summarized in Fig. 4. Reactions identified as *bothside* have a non-carbon imbalance on the reactant and product side. These cases are not solvable by the rule-based method and are hence subject to the MCS-based method. Both methods utilize functions from `RDKit` [44]. Either for parsing reaction SMILES or handling the molecular graph representation in the MCS-based method.

Just like the rule-based method, the MCS-based method can only solve some imbalances. More precisely, the approach depends on the identification of the chemically correct MCS. The method outlined above, in particular, cannot handle rearrangement reactions or ring-formations. We shall return to this point in more detail, see Section 3.2 below. The MCS-based method also tends to fail if too many compounds or boundaries are found, the number of boundaries does not match, or the reaction is not carbon balanced afterwards, e.g., because not all carbon atoms in Y are covered by MCS matches. On the other hand, if a solution is found, the confidence is high that the result is in fact correct.

2.5 Datasets and Benchmarking

SynRBL is not trained on any specific dataset but leverages basic chemical knowledge to inform its rule set. In order to assess its performance we use three widely used public data collections: (i) an open-access tailored for CASP that incorporates the Golden dataset [27], (ii) Jaworski's dataset [19], and (iii) the USPTO_50k collection [5]. The latter contains more than 50,000 reactions. We extracted a representative subset comprising only unbalanced reactions and selected validation datasets based on three different strategies, resulting in the following three datasets. The *USPTO Random Class* dataset (Urnd) was chosen utilizing a stratified sampling method across ten varied chemical reaction classes. Additionally, the *USPTO Different* dataset (Udiff) was selected employing a similar stratified strategy, albeit with Δ, the difference in the dictionaries representing reactants and products, to ensure a comprehensive representation of the diversity in molecular formulas between reactants and products. The *USPTO Unbalance Class* (Uunb) was selected by randomly choosing from reactions classified as solved or unsolved by the rule-based method. This selection provides insights into carbon and non-carbon imbalances within the chosen reaction classes. To ensure reproducibility, the random seed was set to a fixed value (seed value = 42) for all random selection processes. The datasets are summarized in Table 2.

Table 2: Composition of validation datasets in different categories

Dataset	**Reactions**	**C_{unb}**	**Balance**	**Unbalance**
Golden	1851	729	209	913
Jaworski	637	116	302	219
Urnd	803	328	0	475
Udiff	1589	355	0	1234
Uunb	540	257	0	283
Total	5420	1785	511	3124

In order to benchmark SynRBL we evaluated (1) *success* of the algorithm, defined as the fraction of (unbalanced) instances for which SynRBL proposed a balanced reaction, and (2) *accuracy*, the fraction of proposed solutions for the rebalancing problem that are (chemically) correct.

2.6 Estimating Prediction Confidence

The results for the five datasets mentioned in Table 2 were checked manually by TLP, the first author, an experienced chemist. We reviewed all reactions to determine their chemical validity, typically focusing on whether the reaction center or bond changes were valid. The results presented in Section 3 provide a good indicator of how many of the imputations should be correct. However, validating individual outcomes necessitates the expertise of a domain specialist. Predicting a confidence for results from the MCS-based method can be used to filter out potentially wrong imputations and increase the accuracy of the method. We observed that the accuracy strongly depends on the complexity of the reaction center, for example on the number of bonds involved in the reaction. We therefore developed a machine learning model using the `XGBoost` algorithm [48] (version 2.0.3) to predict a confidence value for our imputations based on the reaction properties illustrated in Table 3. This model was trained on 80% of the 2275 reactions from the five datasets that are subject to the MCS-based method, and the remaining 20% (455) of reactions are used for testing.

Table 3: Features for analysis.

Features	Description
total_carbons	The total count of carbon atoms present in the reactions.
total_bonds	The aggregate number of chemical bonds in the reactions.
total_rings	The total count of ring structures within the reactions.
fragment_count	The total number of distinct fragments or molecules present in the reactions.
carbon_difference	The discrepancy in the number of carbon atoms between reactants and products.
num_boundary	The count of boundary atom (reaction center) identified by MCS-based method.
Bond Changes	The maximum count of bonds formed in products or broken in reactants, a feature that requires manual extraction.
bond_change_merge	The net change in the number of bonds between reactants and products post-MCS process.
ring_change_merge	The net change in the number of rings between reactants and products post-MCS process.

To optimize the performance of the model in light of the imbalanced dataset, where the number of correct and incorrect solutions varies significantly, we employ the `SMOTETomek` algorithm [49] from `imblearn` 0.12.0 [50]. This technique combines the Synthetic Minority Over-sampling Technique (SMOTE) with Tomek links to effectively balance the dataset, thereby enhancing the predictive accuracy of our model.

3 Results and Discussion

3.1 Rule-based Method

The rule-based approach of Section 2.2 is applicable on the reactions with missing compounds among either the reactants or the products, with the stipulation that the carbon must be balanced. This method yields a good *success* rate ranging from 89.60% to 99.69% on our five benchmarking sets. It reaches a rather remarkable accuracy level of up to 99.91% on the successful instances. These results are summarized in Fig. 6A below.

Analysis of Incorrect Predictions

A careful inspection of invalid imputations revealed some systematic problems associated with specific datasets. Applied to data derived from the USPTO database (`Urnd`,

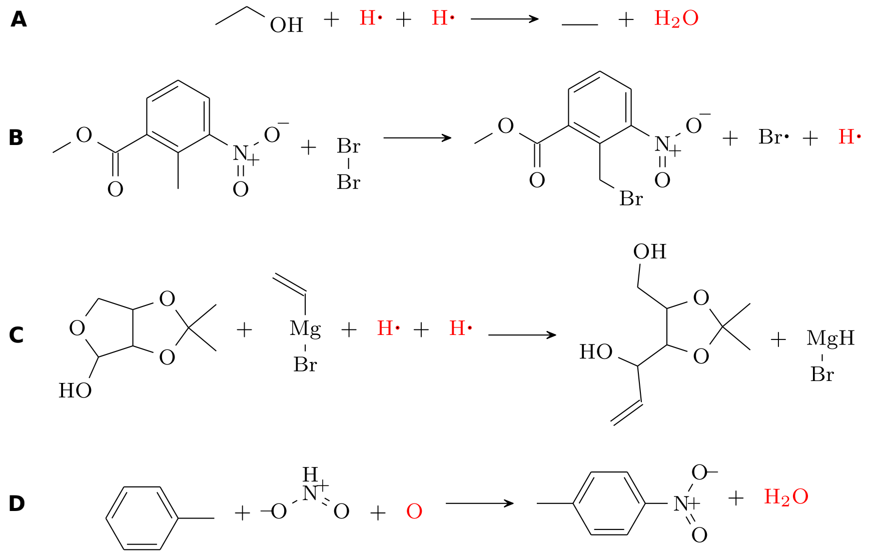

Fig. 5: Examples for incorrect imputations with the rule-based method. Original database entries are shown in black, imputed compounds in red. (A) An erroneous reaction from `USPTO`, with $\Delta = \{O : 1, Q : 0\}$, representing a sequence of dehydration and reduction reactions. (B) A correctly rebalanced reaction from Jaworski dataset that remains uncertain due to the presence of Hydrogen on the product side. (C) False imputation in Jaworski dataset where the product is mistakenly standardized as RMgH instead of RMg^+. (D) An error in the rebalanced reaction in `Golden` dataset, due to HNO_2 being incorrectly identified instead of HNO_3 on the reactant side.

`Udiff`, `Uunb`) the rule base method produced uncertain predictions associated when $\{O : 1, Q : 0\}$ being on the reactant side during rule application. Consider, for example the conversion of ethanol to ethane in Fig. 5A, which is usually performed by dehydration and subsequent hydrogenation or by application of hydroiodic acid HI.

In the Jaworski dataset, two reactions were flagged as uncertain or invalid. The first instance involved the presence of hydrogen in the product without alkali metals or hydrides. This anomaly was traced back to a precursor reaction involving a bromine radical $Br\cdot$, from which the the generation of a hydrogen radical $H\cdot$ is incorrectly inferred. Instead of separate radicals, the formation of hydrogen bromide HBr is expected, see Fig. 5B. Further scrutiny revealed inaccuracies e.g. in Grignard Reactions, where the product was incorrectly identified as RMgH instead of RMg^+. This error could be attributed to the standardized procedures of the original database, which led to the improper imputation of hydrogen on the reactant side. The appropriate correction would be the addition of H^+ to the reactant side and RMg^+ to the product side, Fig. 5C.

In the `Golden` dataset we found 22 reactions with ambiguous status due to invalid reactants. Notably, the formation of nitrobenzene from benzene (id_481, Fig. 5D), erroneously specified nitrous acid HNO_2 instead of nitric acid HNO_3 as the reagent.

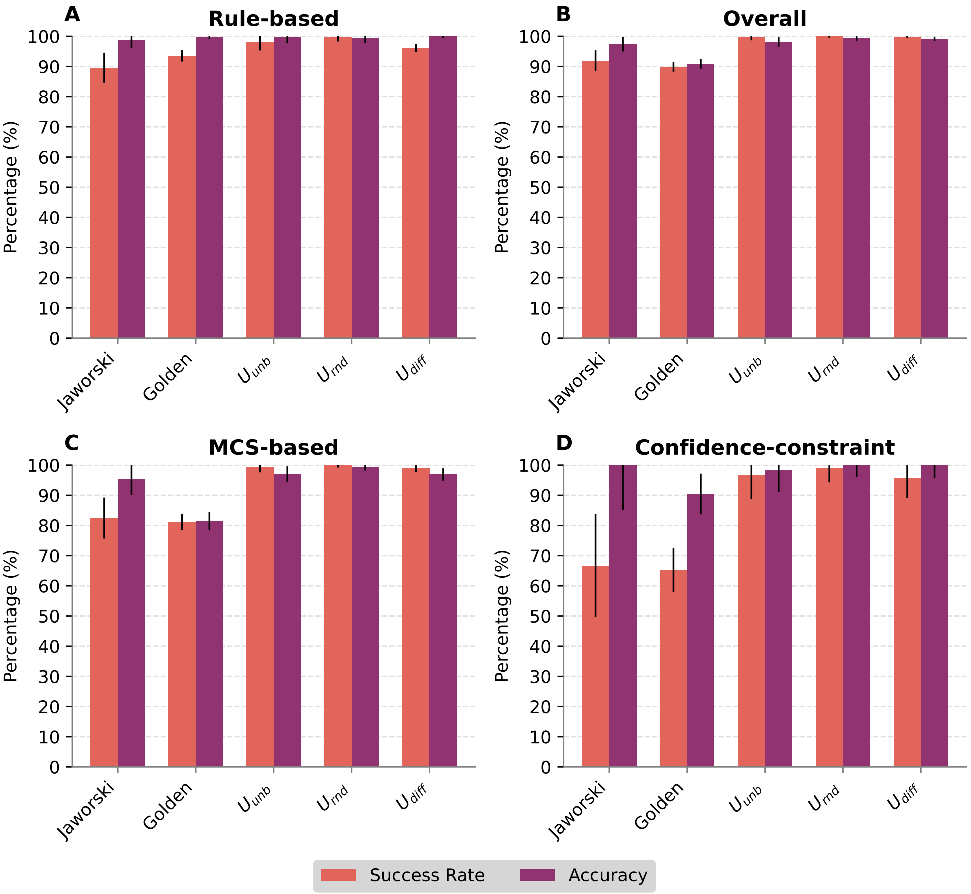

Fig. 6: Validation results for the rule-based method (A), the entire framework (B), MCS-based method (C), and the MCS-based method with an applied confidence threshold of 50% (D). Comparing (C) and (D) shows the tradeoff in success rate for higher accuracy when thresholding the predicted confidence. Because validation was only done on data that was not used in training (20% of the data), (D) has noticeably larger uncertainty margins.

The invalid reactions are enumerated in a dedicated supplementary file. A recurrent pattern observed in these reactions is that the rule-based method infers a singular oxygen O to be added to the reactant side.

Overall, however, the rule-based method rarely produces chemically incorrect or questionable imputations, at least when reactants and products are chemically accurate. The presence of isolated O or H in the prediction, on the other hand, appears to serve as an indicator for errors in the database entry.

The rule-based approach is challenging with respect to computational cost if the compounds contain a larger number of carbon atoms and, in particular, if the number of carbon isomers becomes large. We also note that the method has difficulties with carbon-imbalaneced compounds in general. For example, in the reaction $CH_3COOC_2H_5 \longrightarrow CH_3COOH$, a naive solution might suggest adding ethylene C_2H_4 to balance the product side. The correct solutions, however, is to add water H_2O to the reactants and ethanol C_2H_5OH to the products. Since such examples are abundant, we do not apply the rule-based method to carbon-imbalanced reactions.

3.2 MCS-based Method

The MCS-based method succeeds in 81% (`Golden` dataset) to 100% (`Urnd` of the test cases, see Fig. 6C and Supplementary Table S4. Fig. 7 depicts some reactions that were successfully balanced by the MCS-based method. It showcases the application of a list of different expand and merge rules. In contrast to the rule-based approach, the prediction accuracy on successful cases is not fully satisfactory on all test sets. While the predictions are close to perfect on the USPTO-based datasets, and about 95% for the Jaworski's data, only about 80% are achieved on the `Golden` set. The differences in success rates between the datasets can be attributed primarily to differences in the frequency of reactions that cannot be balanced by the MCS-based approach, in particular rearrangement reactions, ring-formations, or complex reactions with many compounds.

Analysis of Incorrect Predictions

Incorrect predictions arise in particular for complex reactions, and especially with multi-step reactions. Fig. 8 illustrated examples of a ring-forming reaction and a rearrangement reaction where the MCS-based approach fails to identify a valid solution. The structure highlighted as the MCS search result, particularly in Fig. 8B, exhibits four boundaries, indicating an erroneous outcome from the MCS-based method. Such reactions, not amendable by this method, are left unbalanced and represent a limitation of our approach in its current form.

In order to better understand other factors contributing to incorrect predictions, we investigated the influence of different features on the accuracy—see also Section 2.6. Not surprisingly, the accuracy decreases with indicators for the "complexity" of the reaction, particularly with the inferred number of broken/formed bonds, the total number of substances in the reaction, and the number of boundaries. A similar trend is found for the number of different bonds and cycles after graph merging. In contrast, the performance does not depend systematically on the carbon imbalance $|n_C^+ - n_C^-|$. The total number of compounds in a reaction exceeds 6 only in some entries in the `Golden` dataset since it also reports catalysts and solvents. This suggests that the performance declines with more fragments due to potential substance-matching misalignments. In some cases, no boundaries were detected in the MCS step. The lack of accuracy in the absence of a boundary strongly suggests to exit without success if no boundary is found, since the result is almost always wrong anyway. The details of this exploratory data analysis are summarized in Supplementary Fig. S2.

In order to understand the factors influencing accuracy in more detail we performed a feature importance study summarized in Fig. 9A. The feature importance is the average gain, i. e. the relative contribution of each feature for a given prediction over

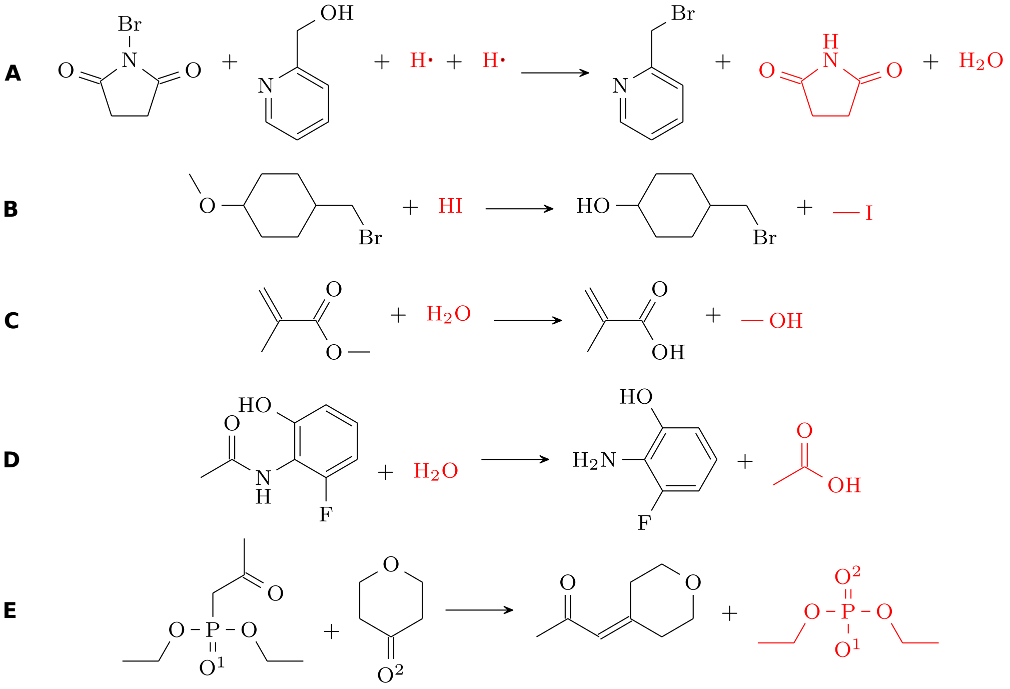

Fig. 7: Some examples of reactions solved by the MCS-based method showcasing different merge and expand rules. Data base entries are shown in black, imputed compounds in red. (A) Append compounds without forming a bond. (B) Append and merge I on Ether break. (C) Append and merge O on Ether break. (D) Append and merge O on Amide break. (E) Create new double bond with P. The double bond between O^1 and P in the reactant is changed to a single bond in the product and the oxygen O_2 from the oxan-4-one creates a double bond with P.

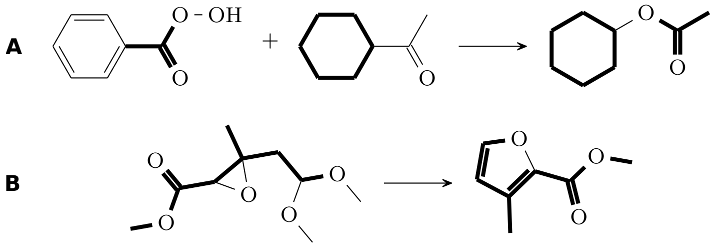

Fig. 8: Two examples that are not solvable by the MCS-based method. The MCS is not meaningful for these types of reactions. (A) Example for an unsolvable oxidation and rearrangement reaction. (B) Example for an unsolvable ring forming reaction. Bold lines indicate the identified MCS.

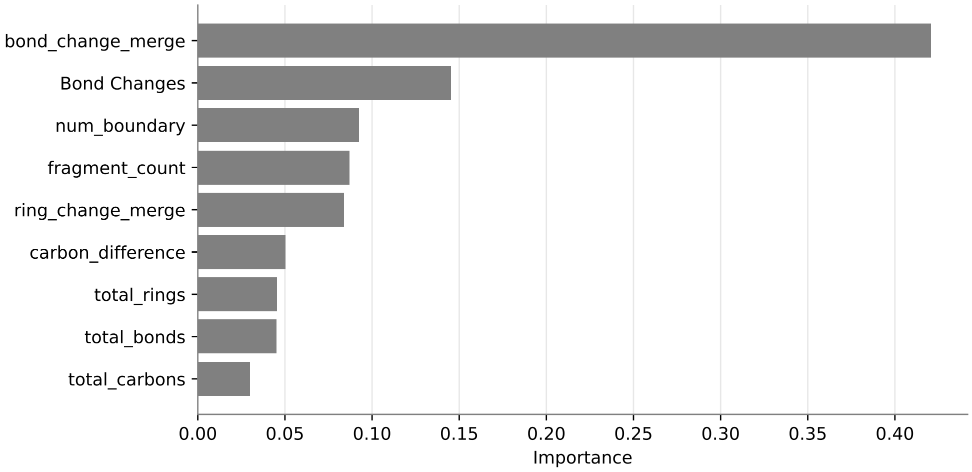

Fig. 9: Feature importance analysis provides a detailed visualization of various factors influencing the precision of the MCS-based method.

.

all targets. In line with the exploratory analysis described above, we observed that the total number of carbons, bonds, rings, and the difference in carbon content within the reaction does not significantly influence the performance of `SynRBL`. Surprisingly, the disparity in the bond count after graph merging emerged as the most impact factor, surpassing even the number of bond changes in predictive power. In order to investigate the interplay between the most informative factors, we also considered the co-occurances of the number of different bonds after merging, the number of different rings after merging, the count of boundaries detected, and total number of compounds, see Fig. S3.

Taken together, this analysis establishes parameters for which we can expect reliable rebalancing results: bond changes after merging should not exceed three; ring changes should be fewer than two; reaction not involve more than four molecules, and only one or two boundaries should be detected.

As a more quantitative approach, we devised a scoring function that summarizes the feature analysis and allows to estimate the confidence level of our predictions, see Section 2.6. The performance of our model is detailed in Supplemental Fig. S4, showcasing strong predictive capabilities with an F1-score (micro) of 0.92, an AUC of 0.94, and an AP of 0.81. Using a confidence threshold of 50%, leads to the expected increase in accuracy of the MCS-based predictions for both Jaworski's dataset and the `Golden` dataset, at a moderate decline in success rate, see Fig. 6D. This observation underscores the robustness of the method in enhancing prediction reliability through the strategic application of a confidence threshold.

3.3 Performance of the Combination of Rule-base and MCS-base Components

The interplay of the rule-based and MCS-based methods described in Section 2.4 results in a satisfactory performance of the SynRBL framework. Fig. 6C shows that the tool reaches success rates between 89.8% (Golden) and 100% (Urnd) at accuracies between 90.8% (Golden) to 99.4% (Urnd). More detailed values are listed in Supplementary Table S4. The significantly lower performance metrics observed within the Golden dataset can be attributed to the inherent complexity of the its reactions, which also include the presence of solvents and catalysts. These elements introduced additional variables into the molecular alignment process, thus posing significant challenges to the predictive capabilities of this framework. In addition, we evaluated the computational efficiency of our methods, observing an average processing time of 46 seconds per 1000 reactions on an average workstation where one-third of the reactions were solved by MCS. In our comparative analysis, our method surpassed the current state-of-the-art, ChemMLM [29], demonstrating superior performance in both *success rate* and *accuracy*. The reported outcomes for ChemMLM showed a *success rate* fluctuating between 4.1% to 42.7% on the USPTO dataset. In contrast, SynRBL demonstrates a remarkable *success rate* of 99% or higher on the same dataset. Moreover, while the *accuracy* of ChemMLM varied widely (from 100% for shorter SMILES strings to a mere 8.2% for larger molecules). SynRBL's accuracy remains robust, largely unaffected by molecular size, and consistently exceeds 98% across the USPTO dataset.

4 Conclusion

In this contribution, we investigated the SynRBL framework as an innovative approach for the rebalance of incomplete reaction entries in chemical databases. SynRBL combines a rule-based approach for carbon-balanced reactions and the MCS-based workflow for carbon-unbalanced reactions. The latter combines variants of the MCIS and MCES problem to increase the fraction of instances in which chemically correct subgraph embedding is found. For the MCS-based component, moreover, a trained feature-based machine learning model was used to estimate the prediction confidence. SynRBL was rigorously evaluated based on five meticulously curated validation datasets, encompassing a subset of the Golden dataset, the Jaworski dataset, and three variants of the USPTO 50k database. Overall, the framework achieves unprecedented accuracy, exceeding 99% on the subset of database entries that it can process successfully. These cover more than 90% of the unbalanced reactions in the datasets used for evaluation. As a by-product of the rule-based analysis, we observed that the signature $O : 1, Q : 0$ referring to a single oxygen is as a strong indication for an error in database entry.

The current implementation of SynRBL is limited to product-dominated or reactant-dominant reaction entries. Moreover, it does not cover certain types of carbon-unbalanced reactions, in particular cyclizations and other complex rearrangement reactions that are difficult for the MCS-based branch of the framework. The SynRBL software is designed, however, to facilitate future extensions of the rule sets as well as of the MCS strategies. SynRBL is not based on a machine learning approach.

Instead, it makes use of “textbook-level” knowledge of chemical reactions in combination with conceptually simple optimization problems. While it does not cover all situations and hence leaves a few percent of the database entries unbalanced, this approach has the advantage of being independent of specific training data and thus of biases inherent in specific data sources. We observed that it indeed yields robust results for datasets with very different chemical content.

Reaction rebalancing with SynRBL can provide much larger and more diverse sets of stoichiometrically balanced reactions as a basis for a wide variety of data-driven tasks in cheminformatics. In particular, we expect that better atom-atom-maps can be obtained from such balanced data since the mappers are freed from the need to solve the reaction balancing problem simultaneously. We expect beneficial effects also on learning approaches, e.g. in forward prediction, retrosynthesis planning, and, notably, the elucidation of reaction mechanisms. Finally, representations of reaction mechanisms in the form of graph transformation rules [51] could be employed as an orthogonal validation strategy, particularly on data sources where *named reactions* are annotated in the metadata.

5 Availability of Data and Materials

The datasets supporting the conclusions of this article are available in the SynRBL repository: https://github.com/TieuLongPhan/SynRBL/tree/main/Data. The source code is avaiable at: https://github.com/TieuLongPhan/SynRBL.

6 Acknowledgement

This project has received funding from the European Union’s Horizon 2021 research and innovation program under the Marie Skłodowska-Curie grant agreement No 101072930 (TACsy – Training Alliance for Computational systems chemistry).

References

[1] Lowe DM (2012) Extraction of chemical structures and reactions from the literature. Tech. rep., Apollo – University of Cambridge Repository, DOI 10.17863/CAM.16293

[2] Goodman J (2009) Computer software review: Reaxys. Journal of Chemical Information and Modeling 49(12):2897–2898, DOI 10.1021/ci900437n

[3] Segler MH, Preuss M, Waller MP (2018) Planning chemical syntheses with deep neural networks and symbolic AI. Nature 555(7698):604–610, DOI 10.1038/nature25978

[4] Schreck JS, Coley CW, Bishop KJ (2019) Learning retrosynthetic planning through simulated experience. ACS central science 5(6):970–981, DOI 10.1021/acscentsci.9b00055

[5] Liu B, Ramsundar B, Kawthekar P, Shi J, Gomes J, Luu Nguyen Q, Ho S, Sloane J, Wender P, Pande V (2017) Retrosynthetic reaction prediction using neural sequence-to-sequence models. ACS central science 3(10):1103–1113, DOI 10.1021/acscentsci.7b00303

[6] Schwaller P, Petraglia R, Zullo V, Nair VH, Haeuselmann RA, Pisoni R, Bekas C, Iuliano A, Laino T (2020) Predicting retrosynthetic pathways using transformer-based models and a hyper-graph exploration strategy. Chemical science 11(12):3316–3325, DOI 10.1039/C9SC05704H

[7] Coley CW, Rogers L, Green WH, Jensen KF (2017) Computer-assisted retrosynthesis based on molecular similarity. ACS central science 3(12):1237–1245, DOI 10.1021/acscentsci.7b00355

[8] Coley CW, Thomas III DA, Lummiss JA, Jaworski JN, Breen CP, Schultz V, Hart T, Fishman JS, Rogers L, Gao H, et al (2019) A robotic platform for flow synthesis of organic compounds informed by AI planning. Science 365(6453):eaax1566, DOI 10.1126/science.aax1566

[9] Gao H, Struble TJ, Coley CW, Wang Y, Green WH, Jensen KF (2018) Using machine learning to predict suitable conditions for organic reactions. ACS central science 4(11):1465–1476, DOI 10.1021/acscentsci.8b00357

[10] Schneider N, Lowe DM, Sayle RA, Tarselli MA, Landrum GA (2016) Big data from pharmaceutical patents: a computational analysis of medicinal chemists' bread and butter. Journal of medicinal chemistry 59(9):4385–4402, DOI 10.1021/acs.jmedchem.6b00153

[11] Coley CW, Barzilay R, Jaakkola TS, Green WH, Jensen KF (2017) Prediction of organic reaction outcomes using machine learning. ACS central science 3(5):434–443, DOI 10.1021/acscentsci.7b00064

[12] Schwaller P, Gaudin T, Lanyi D, Bekas C, Laino T (2018) "Found in Translation": predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. Chemical science 9(28):6091–6098, DOI 10.1039/C8SC02339E

[13] Qian WW, Russell NT, Simons CL, Luo Y, Burke MD, Peng J (2020) Integrating deep neural networks and symbolic inference for organic reactivity prediction. ChemRxiv DOI 10.26434/chemrxiv.11659563.v1

[14] Watson IA, Wang J, Nicolaou CA (2019) A retrosynthetic analysis algorithm implementation. Journal of cheminformatics 11(1):1–12, DOI 10.1186/s13321-018-0323-6

[15] Schwaller P, Vaucher AC, Laino T, Reymond JL (2021) Prediction of chemical reaction yields using deep learning. Machine learning: science and technology 2(1):015,016, DOI 10.1088/2632-2153/abc81d

[16] Probst D, Schwaller P, Reymond JL (2022) Reaction classification and yield prediction using the differential reaction fingerprint DRFP. Digital discovery 1(2):91–97, DOI 10.1039/D1DD00006C

[17] Ghiandoni GM, Bodkin MJ, Chen B, Hristozov D, Wallace JE, Webster J, Gillet VJ (2019) Development and application of a data-driven reaction classification model: comparison of an electronic lab notebook and medicinal chemistry literature. Journal of chemical information and modeling 59(10):4167–4187, DOI 10.1021/acs.jcim.9b00537

[18] Schneider N, Lowe DM, Sayle RA, Landrum GA (2015) Development of a novel fingerprint for chemical reactions and its application to large-scale reaction classification and similarity. Journal of chemical information and modeling 55(1):39–53,

DOI 10.1021/acs.jcim.5b00046

[19] Jaworski W, Szymkuć S, Mikulak-Klucznik B, Piecuch K, Klucznik T, Kaźmierowski M, Rydzewski J, Gambin A, Grzybowski BA (2019) Automatic mapping of atoms across both simple and complex chemical reactions. Nature communications 10(1):1434, DOI 10.1038/s41467-019-09440-2

[20] Schwaller P, Hoover B, Reymond JL, Strobelt H, Laino T (2021) Extraction of organic chemistry grammar from unsupervised learning of chemical reactions. Science Advances 7(15):eabe4166, DOI 10.1126/sciadv.abe4166

[21] Liu T, Cao Z, Huang Y, Wan Y, Wu J, Hsieh CY, Hou T, Kang Y (2023) SynCluster: Reaction Type Clustering and Recommendation Framework for Synthesis Planning. JACS Au 3(12):3446–3461, DOI 10.1021/jacsau.3c00607

[22] Strieth-Kalthoff F, Sandfort F, Kühnemund M, Schäfer FR, Kuchen H, Glorius F (2022) Machine learning for chemical reactivity: The importance of failed experiments. Angewandte Chemie International Edition 61(29):e202204,647, DOI 10.1002/anie.202204647

[23] Llanos EJ, Leal W, Luu DH, Jost J, Stadler PF, Restrepo G (2019) The exploration of the chemical space and its three historical regimes. Proc Natl Acad Sci USA 116:12,660–12,665, DOI 10.1073/pnas.1816039116

[24] Hawizy L, Jessop DM, Adams N, Murray-Rust P (2011) ChemicalTagger: A tool for semantic text-mining in chemistry. Journal of cheminformatics 3:1–13, DOI 10.1186/1758-2946-3-17

[25] Jablonka KM, Patiny L, Smit B (2022) Making the collective knowledge of chemistry open and machine actionable. Nature Chemistry 14(4):365–376, DOI 10.1038/s41557-022-00910-7

[26] Nugmanov R, Dyubankova N, Gedich A, Wegner JK (2022) Bidirectional Graphormer for Reactivity Understanding: neural network trained to reaction atom-to-atom mapping task. Journal of Chemical Information and Modeling 62(14):3307–3315, DOI 10.1021/acs.jcim.2c00344

[27] Lin A, Dyubankova N, Madzhidov TI, Nugmanov RI, Verhoeven J, Gimadiev TR, Afonina VA, Ibragimova Z, Rakhimbekova A, Sidorov P, et al (2022) Atom-to-atom mapping: a benchmarking study of popular mapping algorithms and consensus strategies. Molecular Informatics 41(4):2100,138, DOI 10.1002/minf.202100138

[28] Nugmanov RI, Mukhametgaleev RN, Akhmetshin T, Gimadiev TR, Afonina VA, Madzhidov TI, Varnek A (2019) CGRtools: Python library for molecule, reaction, and condensed graph of reaction processing. Journal of chemical information and modeling 59(6):2516–2521, DOI 10.1021/acs.jcim.9b00102

[29] Zhang C, Arun A, Lapkin A (2023) Completing and balancing database excerpted chemical reactions with a hybrid mechanistic-machine learning approach. ChemRxiv DOI 10.26434/chemrxiv-2023-hrgfw

[30] Ehrlich HC, Rarey M (2011) Maximum common subgraph isomorphism algorithms and their applications in molecular science: a review. Wiley Interdisciplinary Reviews: Computational Molecular Science 1(1):68–79, DOI 10.1002/wcms.5

[31] Willett P, Barnard JM, Downs GM (1998) Chemical similarity searching. Journal of chemical information and computer sciences 38(6):983–996, DOI 10.1021/ci9800211

[32] Willett P (2005) Searching techniques for databases of two-and three-dimensional chemical structures. Journal of Medicinal Chemistry 48(13):4183–4199, DOI 10.1021/jm0582165

[33] Stahl M, Mauser H (2005) Database clustering with a combination of fingerprint and maximum common substructure methods. Journal of chemical information and modeling 45(3):542–548, DOI 10.1021/ci050011h

[34] Gardiner EJ, Gillet VJ, Willett P, Cosgrove DA (2007) Representing clusters using a maximum common edge substructure algorithm applied to reduced graphs and molecular graphs. Journal of chemical information and modeling 47(2):354–366, DOI 10.1021/ci600444g

[35] Boecker A (2008) Toward an improved clustering of large data sets using maximum common substructures and topological fingerprints. Journal of chemical information and modeling 48(11):2097–2107, DOI 10.1021/ci8000887

[36] Raymond JW, Watson IA, Mahoui A (2009) Rationalizing lead optimization by associating quantitative relevance with molecular structure modification. Journal of chemical information and modeling 49(8):1952–1962, DOI 10.1021/ci9000426

[37] McGregor JJ, Willett P (1981) Use of a maximum common subgraph algorithm in the automatic identification of ostensible bond changes occurring in chemical reactions. Journal of Chemical Information and Computer Sciences 21(3):137–140, DOI 10.1021/ci00031a005

[38] Fooshee D, Andronico A, Baldi P (2013) ReactionMap: An efficient atom-mapping algorithm for chemical reactions. Journal of chemical information and modeling 53(11):2812–2819, DOI 10.1021/ci400326p

[39] Kawabata T, Nakamura H (2014) 3D flexible alignment using 2D maximum common substructure: dependence of prediction accuracy on target-reference chemical similarity. Journal of chemical information and modeling 54(7):1850–1863, DOI 10.1021/ci500006d

[40] Garey MR, Johnson DS (1979) Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman, San Francisco

[41] Kawabata T (2011) Build-up algorithm for atomic correspondence between chemical structures. Journal of chemical information and modeling 51(8):1775–1787, DOI 10.1021/ci2001023

[42] Pomper P (1962) Lomonosov and the Discovery of the Law of the Conservation of Matter in Chemical Transformations. Ambix 10(3):119–127, DOI 10.1179/amb.1962.10.3.119

[43] Carruthers W, Coldham I (2004) Modern Methods of Organic Synthesis, Cambridge Univ. Press, Cambridge, UK, chap Formation of carbon–carbon single bonds, pp 1–104. DOI 10.1017/CBO9780511811494.003

[44] Landrum G (2013) Rdkit documentation. Release 1(1-79):4

[45] Kozen DC (1992) The design and analysis of algorithms, Springer, chap Depth-first and breadth-first search, pp 19–24. DOI 10.1007/978-1-4612-4400-4_4

[46] Dalke A, Hastings J (2013) FMCS: a novel algorithm for the multiple MCS problem. Journal of cheminformatics 5(Suppl 1):O6, DOI 10.1186/1758-2946-5-S1-O6

[47] Raymond JW, Gardiner EJ, Willett P (2002) Rascal: Calculation of graph similarity using maximum common edge subgraphs. The Computer Journal 45(6):631–644, DOI 10.1093/comjnl/45.6.631

[48] Chen T, He T, Benesty M, Khotilovich V, Tang Y, Cho H, Chen K, Mitchell R, Cano I, Zhou T, et al (2015) Xgboost: extreme gradient boosting. R package version 04-2 1(4):1–4

[49] Batista GE, Prati RC, Monard MC (2004) A study of the behavior of several methods for balancing machine learning training data. ACM SIGKDD explorations newsletter 6(1):20–29, DOI /10.1145/1007730.1007735

[50] Lemaître G, Nogueira F, Aridas CK (2017) Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. J Machine Learning Res 18(17):1–5

[51] Andersen JL, Flamm C, Merkle D, Stadler PF (2016) A Software Package for Chemically Inspired Graph Transformation. In: Echahed R, Minas M (eds) Graph Transformation, ICGT 2016, Springer Verlag, Berlin, Heidelberg, D, Lecture Notes Comp. Sci., vol 9761, pp 73–88, DOI 10.1007/978-3-319-40530-8_5

Supplementary Information

Comparison of the MCS Variants

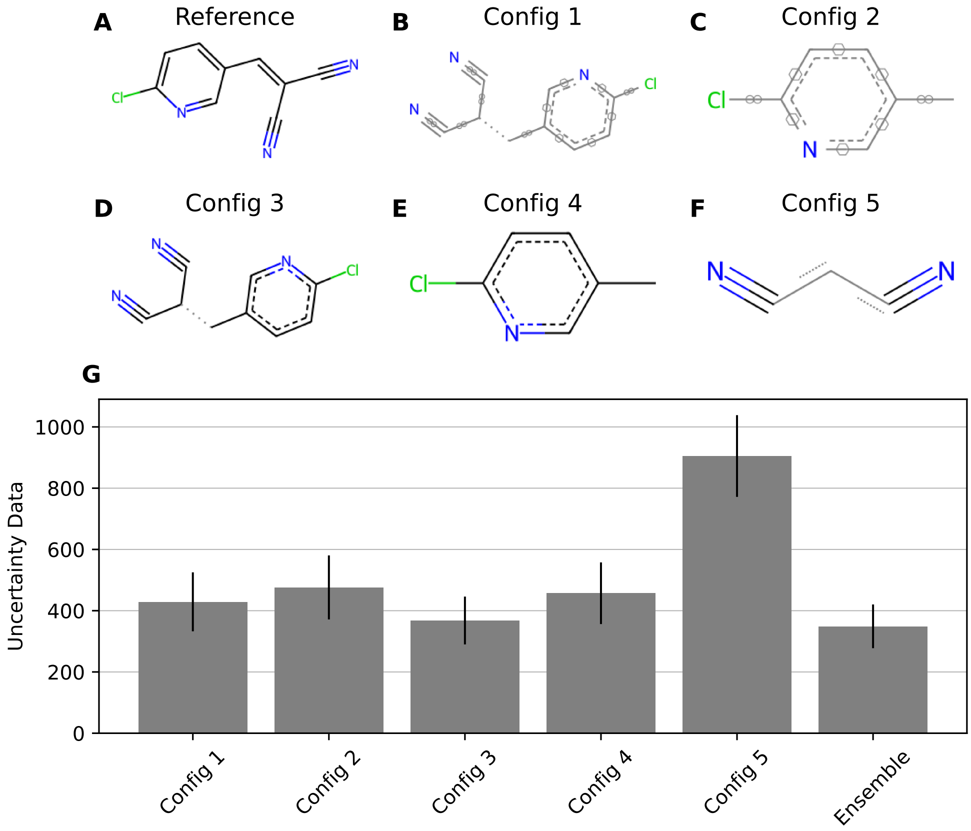

Fig. S1: Benchmarking analysis of MCS search configuration. (A) represents the reference molecules. (B-F) illustrate the MCS results from various configurations. (G) demonstrates the comparative analysis among different configurations and an ensemble method.

As described in the main text, the MCS problem was solved in several different versions ("configurations"), none of which is guaranteed to always identify the chemically correct common subgraph. We benchmarked the different variants and found that they are at least in part complementary. As depicted in Fig. S1, spanning panels A through F, three distinct cases of MCS were identified, where configurations 1 to 4 were MCIS, while configuration 5 was MCES. Notably, the MCES approach demonstrated a capability to expedite the resolution of the NP-hard subgraph isomorphism problem more efficiently than its MCIS counterpart. However, its performance efficacy was suboptimal, a trend observable in Fig. S1G. This discrepancy is likely due to the significant role of bond modifications in chemical reactions, highlighting the dependence of the MCES search on bond-defined substructures. Remarkably, Configuration 3 achieved superior performance, disregarding bond order and complete rings, excluding comparisons with ensemble methods.

These finds emphasize the well-known fact that any particular variant of the graph-theoretical MCS problem does not always identify the chemically correct atom correspondences between molecular graphs. The combination of multiple variations, as implemented in the ensemble method, can achieve at least a moderate improvement, Figure S1G. However, given the additional computational cost of computing multiple MCS solutions, Configuration 3 appears to be best pragmatic choice given its performance and reduced computational requirements. This observation that the ensemble approach improved chemical correctness, albeit slightly, however, can serve as a natural starting point for the development of an improved combinatorial atom-atom-mapping method.

Additional Figures and Tables

Table S1: Merge Rules; FG: Functional Group

Cond. u	Cond. v	Action u	Action v	Bond
O FG: Carbonyl	**P** Pattern: P=O	-	change_bond P=O to P-O	double
O FG: Carbonyl	**P** Pattern: !P=O	-	-	double
O FG: Enol, Alcohol, Phenol	**P**	-	-	single
S	**X**	-	-	no bond
N,O,X	**N,O,X**	-	-	no bond
*	*	-	-	single

Table S2: Expand Rules; FG: Functional Group; cut edge: u - v

Cond. u	Cond. v	FG	Expand
C	O	Ether	I
C	S	Thioether	I
C	O	Ester	O
C	S	Thioester	O
C	N	Amide	O
Mg, Zn, Si, B	*	*	O
O	!O, !N	*	O
N	!O, !N	*	O
C	C	*	O

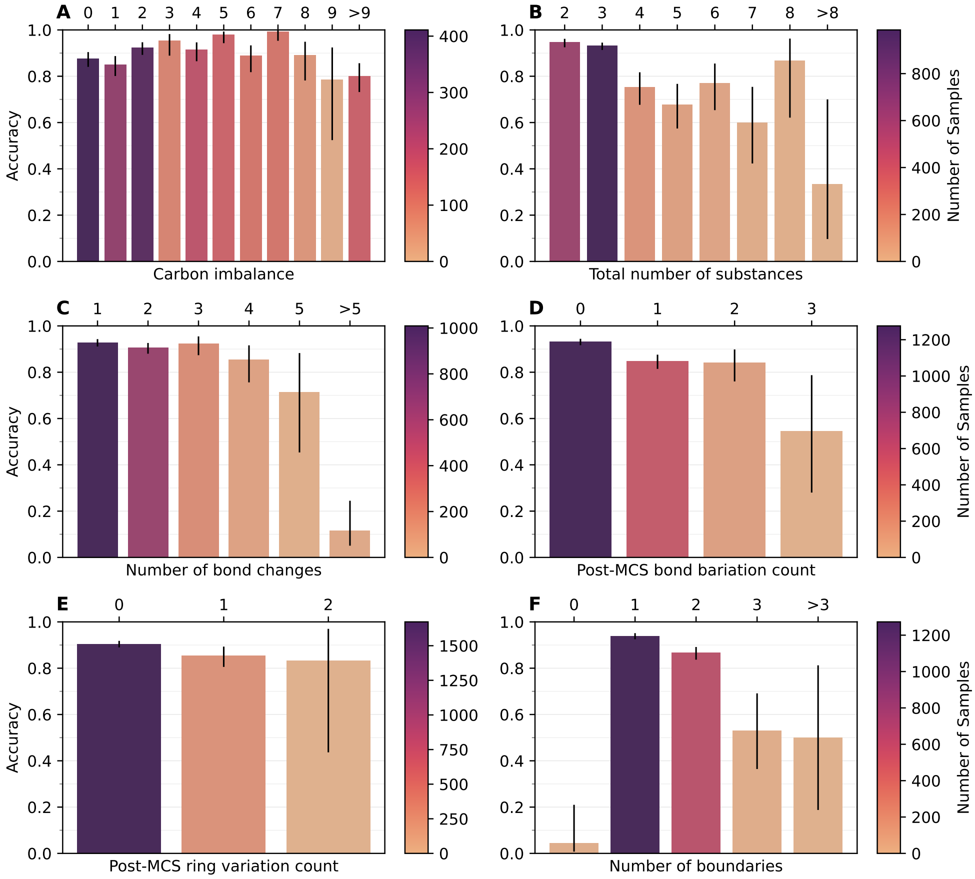

Fig. S2: Exploratory data analysis of MCS-based method performance. (A) Accuracy fluctuates slightly and declines when carbon imbalance exceeds seven. (B) The method performs best with less than four substances. (C) Accuracy drops with over five bond changes, indicating difficulty with rearrangement reactions. (D) Post-MCS bond differences between reactants and products show a decreasing trend similar to bond changes, with optimal performance below three. (E) Ring differences between reactants and products post-MCS show a minor decreasing trend with an increasing number of ring differences. (F) The detection of boundary atoms or reaction centers by MCS is crucial; the method fails without boundary atom detection and underperforms when the number exceeds two.

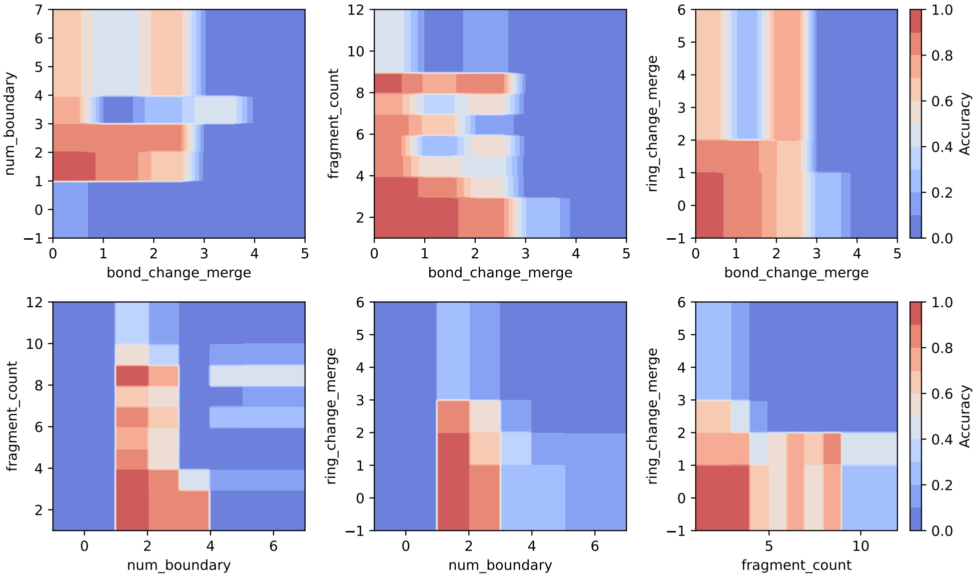

Fig. S3: Contour plots illustrate the confidence region formed by pairs of features. The warm colors in the contour plot represent regions of high confidence, indicating areas where our method demonstrates high accuracy. Conversely, the cool colors denote regions of lower confidence, reflecting areas where our method's accuracy is comparatively lower.

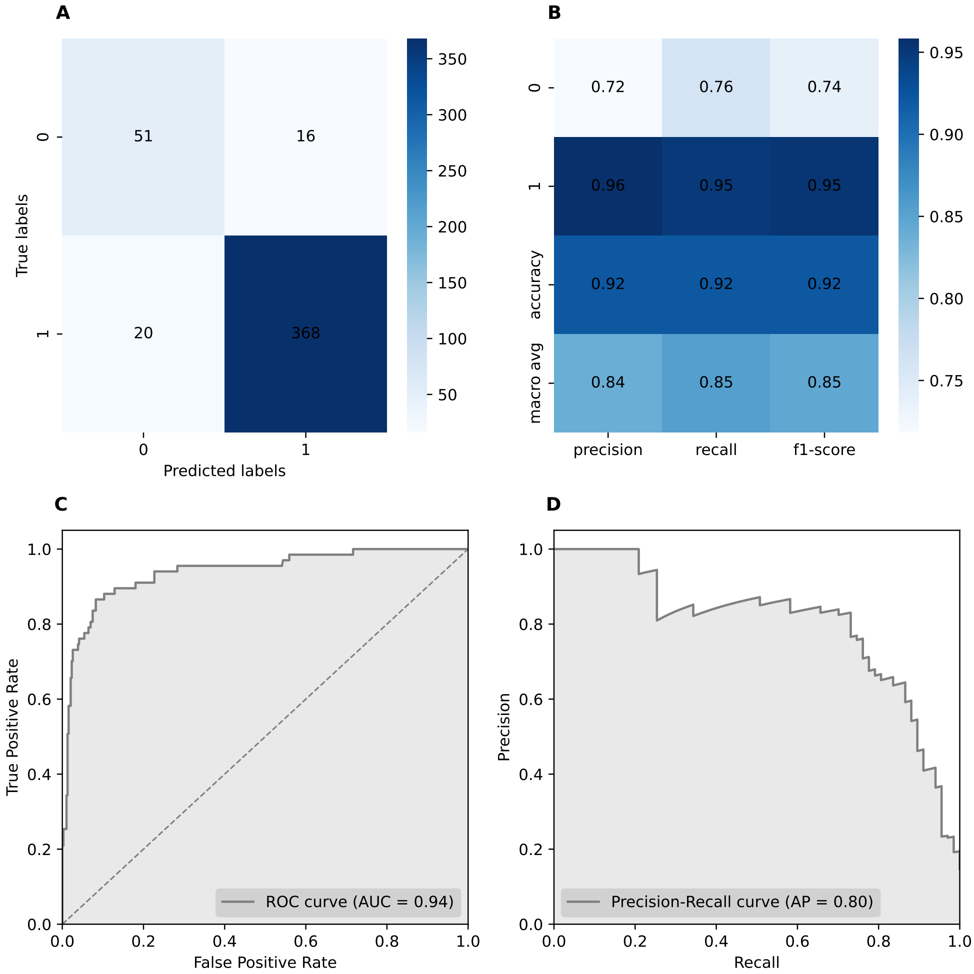

Fig. S4: Evaluation of model performance for a confidence level model using `XGBoost` and `SMOTETomek`. (A) The confusion matrix shows the number of actual versus predicted values. (B) The classification report provides performance metrics, including an F1 score of 0.91. (C) The ROC curve is presented with an AUC of 0.94. (D) The precision-recall curve is shown, with an average precision of 0.8.

Table S3: Library of substitution rules $\hat{r} \rightsquigarrow X_r$ for Section 2.2.2

X_r		$\hat{r}$
Formula	**SMILES**	**Composition**
O	[O]	{O: 1, Q: 0}
Cl_2	ClCl	{Cl: 2, Q: 0}
N_3^-	[N-]=[N+]=[N-]	{N: 3, Q: -1}
H	[H]	{H: 1, Q: 0}
F_2	FF	{F: 2, Q: 0}
Cl_2	ClCl	{Cl: 2, Q: 0}
Br_2	BrBr	{Br: 2, Q: 0}
I_2	II	{I: 2, Q: 0}
H^+	[H+]	{H: 1, Q: 1}
Na^+	[Na+]	{Na: 1, Q: 1}
Li^+	[Li+]	{Li: 1, Q: 1}
K^+	[K+]	{K: 1, Q: 1}
Ca^{2+}	[Ca+2]	{Ca: 1, Q: 2}
Mg^{2+}	[Mg+2]	{Mg: 1, Q: 2}
Ba^{2+}	[Ba+2]	{Ba: 1, Q: 2}
Al^{3+}	[Al+3]	{Al: 1, Q: 3}
Zn^{2+}	[Zn+2]	{Zn: 1, Q: 2}
Cu^{2+}	[Cu+2]	{Cu: 1, Q: 2}
Cu^+	[Cu+]	{Cu: 1, Q: 1}
F^-	[F-]	{F: 1, Q: -1}
Cl^-	[Cl-]	{Cl: 1, Q: -1}
Br^-	[Br-]	{Br: 1, Q: -1}
I^-	[I-]	{I: 1, Q: -1}
N_2	N#N	{N: 2, Q: 0}
O_2	O=O	{O: 2, Q: 0}
S^{2-}	[S-2]	{S: 1, Q: -2}
H_3N	N	{N: 1, H: 3, Q: 0}
H_2O	O	{O: 1, H: 2, Q: 0}
H_2O_2	OO	{O: 2, H: 2, Q: 0}
H_4N^+	[NH4+]	{N: 1, H: 4, Q: 1}
OH^-	[OH-]	{O: 1, H: 1, Q: -1}
NH_3	N	{N: 1, H: 3, Q: 0}
NO_2^-	O=N[O-]	{N: 1, O: 2, Q: -1}
NO_3^-	[N+](=O)([O-])[O-]	{N: 1, O: 3, Q: -1}
NH_2^-	[NH2-]	{N: 1, H: 2, Q: -1}
SO_4^{2-}	[O-]S(=O)(=O)[O-]	{S: 1, O: 4, Q: -2}
PO_4^{3-}	[O-]P(=O)([O-])[O-]	{P: 1, O: 4, Q: -3}
SO_3^{2-}	[O-]S(=O)[O-]	{S: 1, O: 3, Q: -2}
IO_3^-	[O-]I(=O)=O	{I: 1, O: 3, Q: -1}
H_3NO	NO	{N: 1, O: 1, H: 3, Q: 0}
H_4NO^+	[NH3+]O	{N: 1, O: 1, H: 4, Q: 1}
$B(OH)_3$	B(O)(O)O	{B: 1, O: 3, H: 3, Q: 0}
H_3BO_2	B(O)(O)	{B: 1, O: 2, H: 3, Q: 0}
CO_2	C=O	{C: 1, O: 2, Q: 0}
$SOCl_2$	O=S(Cl)Cl	{S: 1, O: 1, Cl: 2, Q: 0}
$H_4N_2O_2S$	NS(N)(=O)=O	{N: 2, S: 1, O: 2, H: 4, Q: 0}
$HClO_3S$	O=S(=O)(O)Cl	{S: 1, O: 3, Cl: 1, H: 1, Q: 0}
$B(OH)_2Cl$	B(O)(O)Cl	{B: 1, O: 2, H: 2, Cl: 1, Q: 0}
$B(OH)_2Br$	B(O)(O)Br	{B: 1, O: 2, H: 2, Br: 1, Q: 0}
$B(OH)_2I$	B(O)(O)I	{B: 1, O: 2, H: 2, I: 1, Q: 0}
H_2ClNO_2S	NS(=O)(=O)Cl	{N: 1, S: 1, O: 2, Cl: 1, H: 2, Q: 0}

Table S4: Comprehensive Performance Metrics of the SynRBL

Dataset	**Jaworski**	Golden	Uunb	Urnd	Udiff
Total number reactions	637	1851	540	803	1589
Number of unbalance reactions	335	1642	540	803	1589
Number of rule solved reactions	181	754	240	324	1134
Rule success rate (%)	89.6	93.55	97.96	99.69	96.1
Number of rule accurate reactions	179	752	239	322	1133
Rule accuracy (%)	98.9	99.73	99.58	99.38	99.91
Number of MCS solved reactions	127	721	298	479	451
MCS success rate (%)	82.47	81.19	99.33	100	99.12
Number of MCS accurate reactions	121	588	289	476	437
MCS accuracy (%)	95.28	81.55	96.98	99.37	96.9
All solved reactions	308	1475	538	803	1585
All success rate (%)	91.94	89.83	99.63	100	99.75
All accurate reactions	300	1340	528	798	1570
All accuracy (%)	97.40	90.85	98.14	99.38	99.05