



ARTICLE

<https://doi.org/10.1038/s41467-019-09440-2>

OPEN

Automatic mapping of atoms across both simple and complex chemical reactions

Wojciech Jaworski¹, Sara Szymkuć², Barbara Mikulak-Klucznik², Krzysztof Piecuch¹, Tomasz Klucznik², Michał Kaźmierowski¹, Jan Rydzewski¹, Anna Gambin¹  & Bartosz A. Grzybowski¹ ^{2,3,4}

Mapping atoms across chemical reactions is important for substructure searches, automatic extraction of reaction rules, identification of metabolic pathways, and more. Unfortunately, the existing mapping algorithms can deal adequately only with relatively simple reactions but not those in which expert chemists would benefit from computer's help. Here we report how a combination of algorithmics and expert chemical knowledge significantly improves the performance of atom mapping, allowing the machine to deal with even the most mechanistically complex chemical and biochemical transformations. The key feature of our approach is the use of few but judiciously chosen reaction templates that are used to generate plausible "intermediate" atom assignments which then guide a graph-theoretical algorithm towards the chemically correct isomorphic mappings. The algorithm performs significantly better than the available state-of-the-art reaction mappers, suggesting its uses in database curation, mechanism assignments, and – above all – machine extraction of reaction rules underlying modern synthesis-planning programs.

¹Faculty of Mathematics, Informatics and Mechanics, University of Warsaw, Ul. Banacha 2, 02-097 Warszawa, Poland. ²Institute of Organic Chemistry, Polish Academy of Sciences, Ul. Kasprzaka 44/52, Warsaw 02-224, Poland. ³IBS Center for Soft and Living Matter, UNIST, 50, UNIST-gil, Eonyang-eup, Ulju-gun, Ulsan, South Korea. ⁴Department of Chemistry, UNIST, 50, UNIST-gil, Eonyang-eup, Ulju-gun, Ulsan, South Korea. These authors contributed equally: Wojciech Jaworski, Sara Szymkuć, Barbara Mikulak-Klucznik. Correspondence and requests for materials should be addressed to A.G. (email: aniag@mimuw.edu.pl) or to B.A.G. (email: nanogrybowski@gmail.com)

Mapping atoms across chemical reactions—that is, numbering them to indicate which atom of the substrate(s) becomes which atom of the product(s)—is not only one of the classic exercises in organic chemistry textbooks^{1,2} but also of growing importance in classifying reactions³ in large databases and facilitating substructure searches⁴, in assigning the roles (reactant, reagent, product) specific molecules play in a given reaction⁵, and in elucidating mechanisms of enzymatic reactions or identifying metabolic pathways^{6–9}. Most recently, atom mapping has become a central component of chemical AI^{10–13} as it is required for automatic extraction of reaction cores/rules from literature precedents; such rules are subsequently used as the knowledge base of various synthesis-design programs^{10–17}. Unfortunately, the existing algorithms^{6–9,18–25} are capable of correctly mapping relatively simple reactions (sometimes only when full stoichiometry is provided²⁵) but not those in which expert chemists would actually benefit from computer's help. In addition, their purportedly high correctness is often reported based on comparisons to the results of other algorithms (i.e., not to the chemically correct mappings²⁵) or to examples from databases in which mappings were never systematically verified^{7–9,22–24}.

Computationally, the atom mapping problem is known to be NP hard (as it encloses the subgraph isomorphism problem²⁶). The existing algorithms for atom matching^{6–9,18–25}—described in comprehensive recent reviews^{18,27}—typically fall into two broad and partly overlapping categories. Methods based on the so-called extended connectivity (EC)^{19,28} are often extensions of the Morgan algorithm²⁹ (which assigns a unique number to each atom in a molecule on the basis of its chemical neighborhood) and use iterative procedures to establish unique labelling of graph vertices (i.e., of atoms), identify common substructure(s) between substrates and products, and ultimately construct complete atom mapping. Another class of algorithms relies on the so-called principle of minimal chemical distance (MCD)³⁰, which is an ansatz stipulating that most chemical reactions follow the shortest path from reactants to products, in the process cutting the minimal possible number of bonds. In order to find the optimal/correct reaction mapping, these algorithms try to solve the subgraph isomorphism problem⁷, use problem solving methods (such as the A*-algorithm)⁸, or introduce integer linear optimization²⁴. In all cases, the algorithms face NP-hard problems which are in general intractable without the use of additional domain knowledge that could reduce the problem's complexity. More significantly, they are incapable of mapping reactions for which the assumptions such as MCD are simply incorrect, as in various types of pericyclic reactions, 1,2-rearrangements, or metathesis reactions for which the number of bonds being cut is not minimal (vs. alternative mappings, see example in Fig. 1a, b). Another problem concerns reactions in which cutting different bonds leads to answers with the same overall algorithm score—one example is the Prins rearrangement in Fig. 1c, d, for which traditional algorithms would not be able to decide whether the oxygen atom in the product's ring comes from substrate 3 (Fig. 1c) or 4 (Fig. 1d). To overcome such problems, Baldi's group²⁵ has recently combined the substructure and optimization methods with an atom-assigning cost function trained on a large (>250,000) set of atom-mapped reactions from the SPRESI database from ICSynth³¹. The authors claimed that for relatively basic reactions with complete stoichiometry (Fig. 1e, f), this approach and the accompanying ReactionMap software³² performed superior to other algorithms (including commercial ChemAxon's software, MarvinJS³³). On the other hand, this work showed one of the major flaws of the field—namely, mapping was considered correct if it matched the mapping of another program (in this particular case, ICSynth's proprietary program used to

map SPRESI reactions). Such examples emphasize the need to validate algorithm's predictions against true mappings performed by human experts and accounting for reaction mechanism.

Here, we describe and extensively validate an algorithm that can map even the most mechanistically complex chemical and biochemical transformations, including those with incomplete stoichiometry. The distinctive feature of our approach is that it supplements graph-theoretical considerations and optimization schemes^{6–9,18–23} with few (20) but judiciously chosen chemical rules/heuristics that allow the algorithm to explore plausible “intermediate” atom assignments, avoid decoy solutions (e.g., those obtained under the assumption of minimal number of bonds being cut^{7–9,22,23}), and ultimately be guided towards the correct mappings. The advantages of this approach are most manifest for complex to very complex reactions where it reaches 84% mapping correctness vs. up to ~62% of other, state-of-the-art solutions^{25,32,33} (and ~86% vs. ~71% over reactions provided with full stoichiometry). We also demonstrate similar improvements in the mapping of reactions from which “synthetic rules” were previously extracted (often incorrectly, as it turns out) and were used as the basis for synthetic design algorithms. Accurate and general-scope mapping algorithms like the one we describe are important to ensure that computers can extract, process, annotate, and apply not only simple chemical transforms but also the advanced chemistries without which any synthesis-design programs cannot address realistic synthetic challenges. The analyses and comparisons described below are based on over 1400 expert-mapped reactions of different complexities (cf. Supplementary Notes 2–5 and file Supplementary Data 1).

Results

Establishing isomorphic mapping. Our algorithm has two major, interrelated components: a module for isomorphic (i.e., one-to-one) mapping and a module for the application of reaction heuristics guiding correct atom assignments. To simplify the problem to the isomorphism of subgraphs (rather than full molecular graphs), we first label atoms by their environments (represented as subgraphs around each atom, not the routinely used scalar values) and use the so-called bucket sorting³⁴ to group together atoms with identical environments. These subsets of atoms within the substrate/product molecules define possible candidates for isomorphic mapping. To reduce solution search space and thus avoid time consuming exhaustive analysis—which is computationally prohibitive for large molecules—we introduce and apply sequentially four combinatorial tests: Test 1: Whether the number of connected components in reactant(s) and product (s) graphs are equal; Test 2: Whether connected components may be matched according to the number of atoms/nodes; Test 3: Whether the connected components may be matched according to node types (i.e., have pairwise identical multisets of nodes' environments); and Test 4: Whether the connected components are pairwise isomorphic. These tests exclude the majority of possible candidates.

The most computationally-intensive part of the above operations is to find the correct isomorphism. This is done by creating a decision tree³⁵ of all possible isomorphisms, with the size of this tree limited to a predefined number of vertices (currently, 1,000,000)—evaluation of a candidate reaction typically requires trees with tens of thousands of vertices but for either very large molecules and/or those with many symmetries, it can approach or even exceed the one-million limit. The procedure is similar to the so-called VF2 algorithm³⁶ with an important difference that VF2 adds single nodes (here, atoms) to the matching while our algorithm extends the matching simultaneously to all immediate neighbors of a given atom. Specifically, the isomorphisms are first

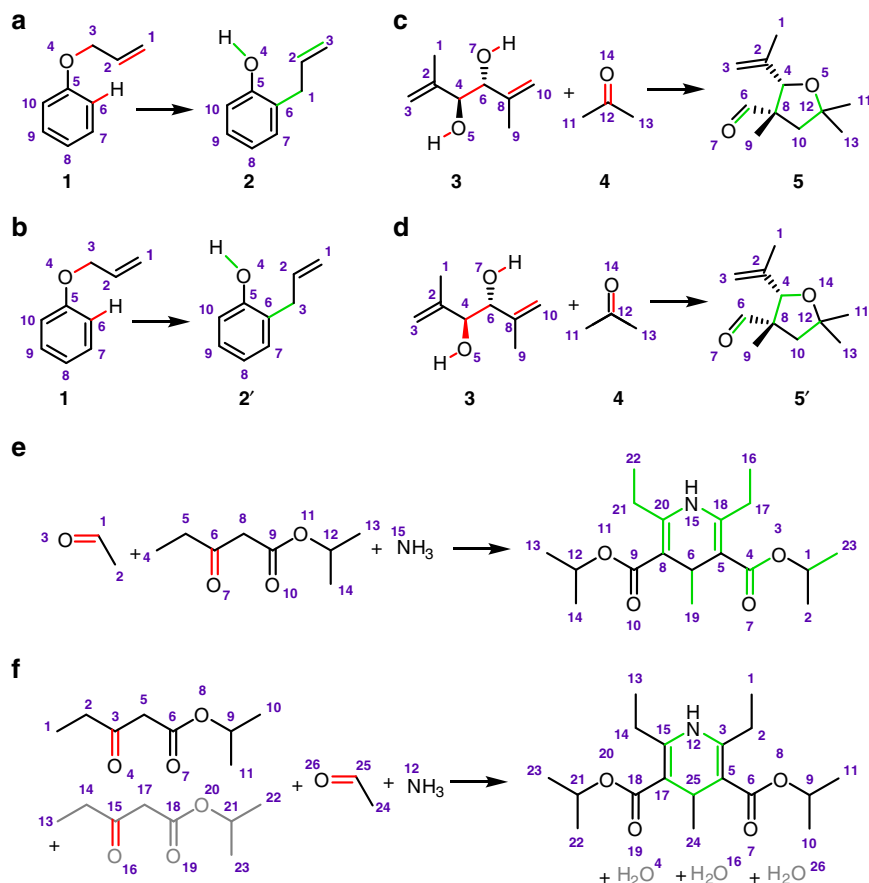
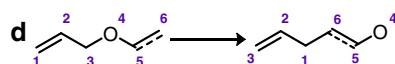
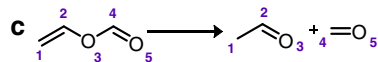
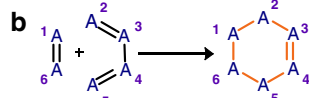
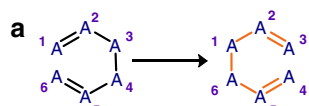


Fig. 1 Examples of problems encountered by traditional matching algorithms. **a, b** The number of bonds being cut might not necessarily be minimal. The example shown here is for a (sigmatropic) Claisen rearrangement¹. The mapping in **(a)** is correct, even though the one in **(b)** entails fewer bonds being cut (bonds that are disconnected are colored in red whereas bonds created are marked green). **c, d** Several alternative mappings with the same “bonds-cut” score might exist. Shown here are two alternative mappings of Prins rearrangement³⁹—in both cases, six bonds are being cut and four are formed. The correct mapping, one determined by our algorithm, is shown in **(c)**. **e, f** When chemists write organic reactions, they usually do not account for full stoichiometry. The example here is for the Hantzsch dihydropyridine synthesis⁴⁰ in which only one molecule of ketoester would typically be provided in a synthetic scheme. **e** Competitive mappers do not account for the missing substrate and either yield incorrect mapping (as shown here based on Marvin³³) or find no mapping at all (Baldi’s ReactionMap³²). **f** Our method considers missing atoms or substrates (here, one with bonds colored gray) and maps the reaction correctly

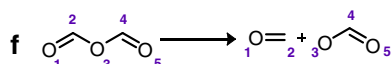
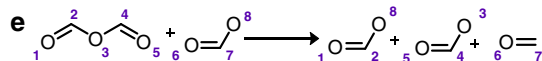
calculated at the level of the fourth-order environments, and then recursively at the third, second, and first levels. The purpose of this recursion is to match the atoms sequentially from the periphery of the molecules (where atoms agree to higher-order environments) towards the reaction center (where the conserved neighborhoods become smaller). Regarding the VF2 part of the algorithm, it checks the graph isomorphism by constructing the matching between graphs’ nodes. For a given partial matching, it computes the set of candidate pairs for inclusion in the matching. Then, for each pair from the candidate set, it executes itself recursively with the matching extended by the selected candidate as an argument. When the algorithm obtains matching that covers all the nodes of one of the graphs, it terminates. When all candidate pairs fail to establish complete matching or the set of the candidate pairs is empty, the algorithm back-traces. If after such procedure some unmatched atoms still remain, we again resort to combinatorics, sequentially cutting/removing all possible subsets of bonds originating from the unlabeled atoms—first all possible single bonds then, if needed, pairs of bonds, and so on up to sets of six bonds. By this bond cutting we strive to identify minimal sets of bonds defining the “reaction center” and whose disconnection gives a full isomorphism between reactants and products (for further details and examples, see Supplementary Notes 1.1 and 1.2).

Addition of reaction heuristics. Although the above algorithm—using an improved representation of neighborhoods and various original combinatorial tests/procedures—is efficient in mapping simple reactions, it fails, for instance, when the number of disconnected bonds is not minimal or when there are multiple different solutions with the same score (see Fig. 1). To take these and other cases into account, we have augmented the algorithm with 20 reaction heuristics listed in Fig. 2. In addition to generating mapping candidates as described previously, the algorithm now tries to apply these heuristics at all possible sites of the reagent(s). In this way a set—sometimes quite large, up to hundreds—of intermediate candidates is created that is then subject to isomorphic mapping against the product(s) as described above. Importantly, whereas the candidates generated without the use of heuristics are scored based on the number of bonds disconnected (+1 for every bond cut, including bonds involving H atoms), the score for the application of heuristics is defined as lower than the actual number of bonds being cut or changed by each heuristics—that is, heuristics are being preferred to allow the algorithm explore solutions with non-minimal numbers of bonds being cut. In other words, the algorithm still strives to find the mapping with minimal score but the heuristics help “channel” the calculations towards chemically viable solutions. The block diagram of

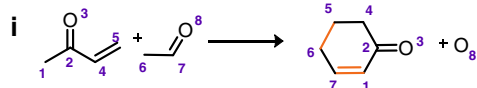
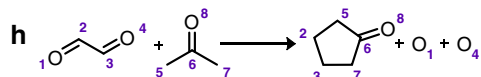
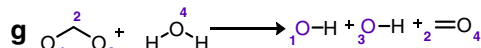
I. Pericyclic reactions heuristics



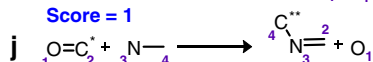
II. Atom distinguishability heuristics



III. Carbonyl group heuristics



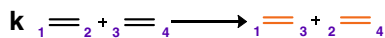
Score = 1



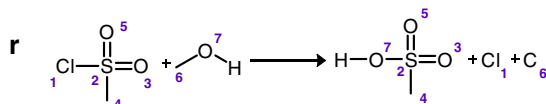
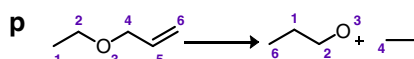
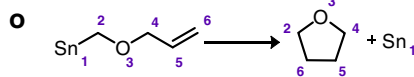
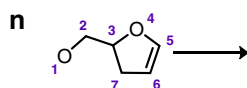
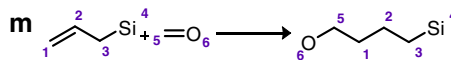
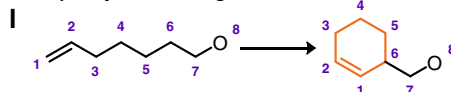
Score = -1.5

* all remaining bonds connected to C₂ are single** a double bond cannot be connected with C₄

IV. Heuristic for "missing" catalyst



V. Non-pericyclic rearrangements heuristics



Score = 0

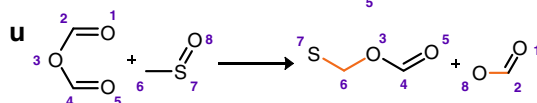
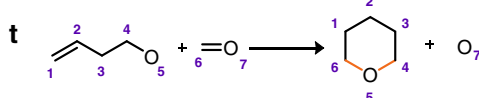
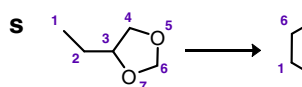


Fig. 2 Schemes of reaction heuristics guiding the searches. Specific heuristics span various reaction classes: **a-d** pericyclic reactions; **e, f** reactions in which two atoms of the same type can be hard to distinguish due to symmetry; **g-j** select reactions involving carbonyl group; **k** reactions such as metathesis; **l-u** non-pericyclic rearrangements. The algorithm strives to apply these templates to the substrates in all possible ways. The intermediates thus created are then subject to the mapping procedure as described in the main text and illustrated in Figs. 3 and 4. The score for the application of a heuristic is usually +0.5 (vs. +1 for each bond cut without the use of heuristics). Exceptions are heuristics **i, j**, and **r**, for which the scores are, respectively, +1, -1.5, and 0. In the templates shown, free valences of all atoms can be filled by atom(s) of any type, unless otherwise stated. Dashed lines specify aromatic bonds; orange color indicates bonds which cannot be broken in any other subsequent operations during mapping

the algorithm is shown in Fig. 3. We make several additional comments about this protocol.

The heuristics can be divided into five sub-groups (see Fig. 2). The first one includes reaction types, to which the ansatz of the minimal number of bonds being cut does not apply (e.g., pericyclic transformations including cycloadditions, electrocyclic reactions, or sigmatropic rearrangements). In the cyclic transition states of such transformations, breaking several bonds is energetically more favorable than cutting one sigma bond (see Fig. 1a, b). Heuristics in the second group are for reactions in which disconnections of certain chemically distinct bonds are indistinguishable from the algorithm's point of view. An example here could be a simple reaction between an anhydride and a carboxylic acid (marked as [e] in Fig. 2) in which another anhydride is formed. In the newly formed anhydride, the central (non-carbonyl) oxygen atom might derive either from the initial anhydride or from carboxylic acid's OH group—both of the solutions are algorithmically equivalent, although only the second variant is chemically correct. The third group

covers chemistry of the carbonyl group (e.g., heuristics [h] and [i] in Fig. 2 suggest that two carbonyl compounds might undergo a condensation, such as aldol, accompanied by the loss of a water molecule). The fourth group of heuristics deals with the problem of incomplete chemical information crucial for the reaction mechanism and outcome but not explicitly present in the substrate(s) or product(s). For example, cutting two double bonds in a metathesis reaction might appear illogical from the algorithm's point of view, because cutting two single bonds instead would result in a lower-score—although chemically incorrect—solution (see Fig. 4b). Chemically, in this case cutting two double bonds makes perfect sense if one accounts for the presence of an organometallic catalyst that coordinates to double bonds and facilitates the bond-breaking step. In other words, this heuristic corrects for the missing catalyst. The fifth and the last group of heuristics comprises important non-pericyclic rearrangements and includes certain reaction motifs popular in chemical transformations of different types.

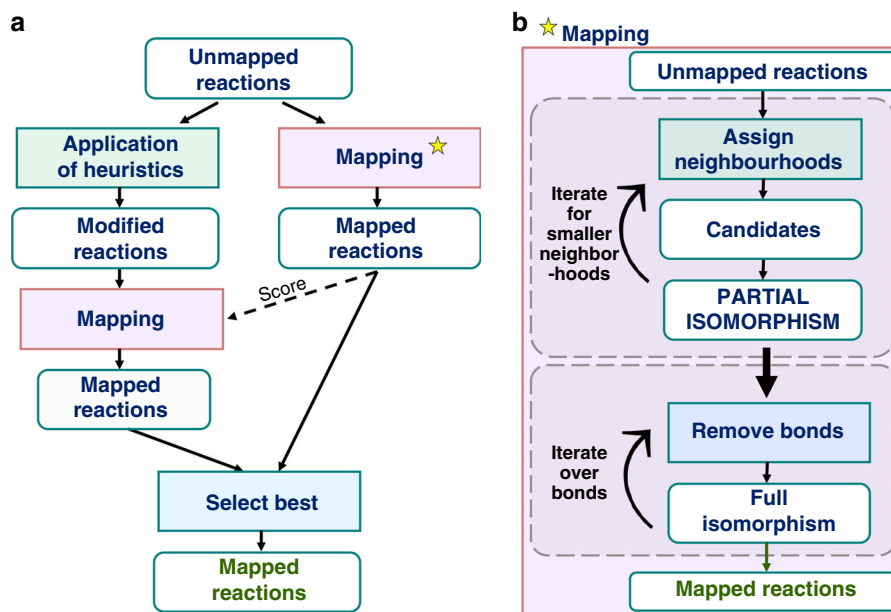


Fig. 3 Algorithm scheme. **a** A diagram illustrating the algorithm with (left arm of the flowchart) and without (right arm) the use of heuristics. Both paths are processed simultaneously—in the end, the scores are compared and a lower-scoring solution is selected. The purpose of additionally performing the search without heuristics is that the best solution it produces (with some score S , communicated from the right to the left branches; dotted arrow) allows rapid rejection of any other with-heuristics solutions for which the score is above S . This allows the algorithm to significantly limit the search space and yield results faster. **b** A scheme of the mapping procedure, in which two iterative processes are involved: iteration for smaller neighborhoods and iteration over bonds—see main text for details

For the best performance of the algorithm, not all rules are assigned the same score preference: the score for the majority of applied heuristics is $+0.5$, but $+1$ for heuristic [i] (Robinson annulation; the value is still lower than if each disconnected bond were scored as $+1$), -1.5 for [j], and 0 for [r].

Any preferences given to the heuristics are substantial only if this heuristics is applied in a proper way. If it is applied at a wrong locus of a molecule, it transforms the substrate into a decoy form (see examples in Fig. 4) that is significantly less similar to the product. Consequently, more bonds need to be cut to obtain the proper product structure (i.e., finding the full isomorphism is more difficult) and the overall score is high.

Addition of any new heuristics not only increases the number of candidates for isomorphic matching and the computation time but can also create more decoy solutions (see Fig. 4) that are chemically improper—the set of 20 heuristics we use was identified by an iterative protocol in which we aimed to minimize the number of heuristics while, without increasing the computation time, maximizing the applicability of the algorithm to chemistries as diverse as possible. When additional heuristics were added, they could rarely help in mapping some specialized type of chemistry but, as a rule, they concomitantly decreased the efficiency of correct mapping of many other reaction types.

Correcting for the missing stoichiometry. The last and important aspect of the algorithm is the ability to match reactions with incomplete stoichiometry. The stoichiometry correction is executed before the main mapping process. The algorithm first checks if the reaction could be balanced by adding copies of some of the substrates (if so, such copies are added; see example in Fig. 1f) or by matching against hard-coded, atom-mapped templates of some popular reactions in which the groups, typically unspecified by chemists while writing reactions, are explicitly included (if a template fits the reaction, the core atoms already have atom assignments and no heuristics are needed, which speeds up the algorithm; for templates, see Supplementary Fig. 6).

In the next stage, the algorithm tries to balance stoichiometry by adding water molecules to either reactants or products. If the reaction is still unbalanced, individual missing atoms are added to the appropriate side of the reaction. The algorithm then treats them as one-atom-molecules and performs full mapping routine as described earlier. This scheme works well for less than ca. eight missing atoms—for larger numbers, the problem becomes intractable due to combinatorial explosion of mapping options.

Discussion

During algorithm development and selection of heuristics (cf. above), we scrutinized its results on the total of 548 reactions which can be referred to as a “training set”. This set comprised 241 typical reactions with full stoichiometry and taken from the Organic Syntheses collection³⁷; set of 191 randomly selected and typically mostly stoichiometrically unbalanced reactions from Reaxys collection³⁸; and 116 mechanistically complex reactions (73 stoichiometrically balanced and 43 unbalanced reactions) taken from various literature sources (e.g., Kurti’s “Strategic Application of Named Reactions in Organic Synthesis”³⁹ or Grossman’s “The Art of Writing Reasonable Organic Reaction Mechanisms”¹)—whose mapping should pose a challenge even to human experts. These reactions were mapped by our algorithm, by ReactionMap³², Marvin JS version 16.4.18³³, and as a benchmark for correctness, by the authors (S.S., B.M.K., T.K., all expert organic chemists with track record as co-developers of the Chematica retrosynthetic software^{13,40}). All mapped reactions are provided in the Supplementary Note 2. For the 241 typical reactions with full stoichiometry, our algorithm provides 93.8% correct assignments compared to 92.1% for ReactionMap and 86.7% for MarvinJS methods—that is, it does slightly better than the competing solutions. For the 191 Reaxys reactions in the second collection, the accuracy of our algorithm is 94.2% vs 90.5% for MarvinJS and only 12% for ReactionMap. The poor performance of ReactionMap could be expected since, as its authors admit²⁵, the program cannot generally tackle reactions

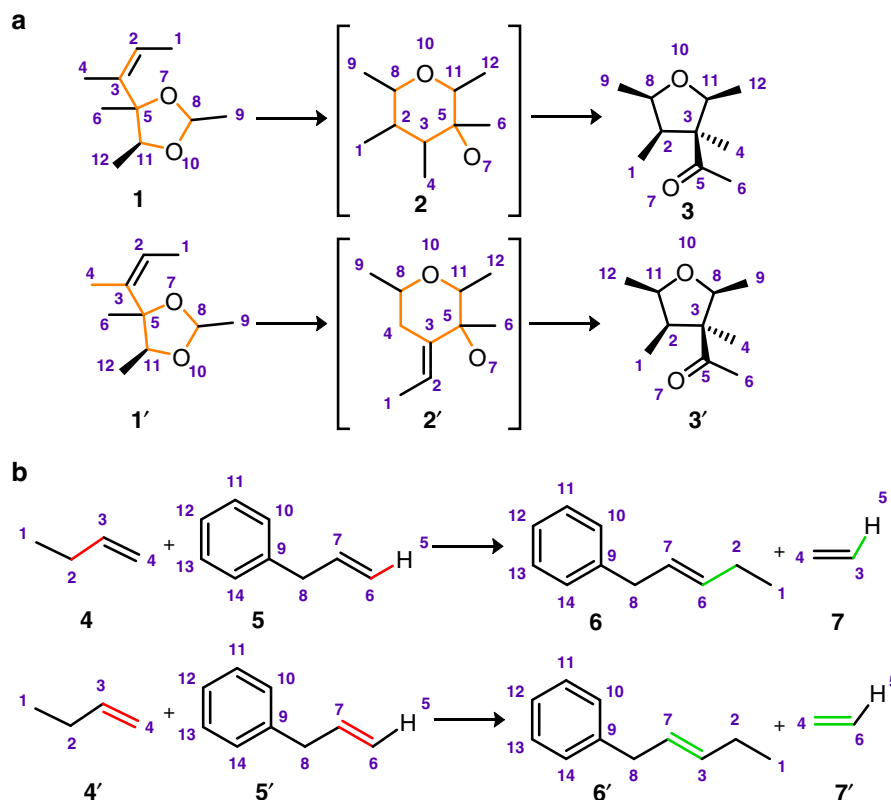


Fig. 4 Application of reaction heuristics. The example in **a**, illustrates the consequences of applying reaction heuristic [s] (see Fig. 2) in various places of the same substrate **1** in the Prins-Pinacol rearrangement^{39,52}. In the top row, the heuristic is properly applied to the substrate (the substructure of the substrate recognized by the heuristic is marked orange) producing intermediate **2**, which is then correctly mapped into product **3** with overall score = 3.5 (3 for disconnecting bonds 5–11, 3–H, and 7–H plus 0.5 for using the heuristic). In contrast, when the same heuristic is applied to the wrong part of the molecule (**1'**), it yields intermediate **2'**, which is incorrectly mapped into product **3'** with score = 4.5 (3 for disconnecting single bonds 4–8, 5–11, 7–H, additional 1 for converting double bond 2–3 into a single bond, and 0.5 for using the heuristic). The second example in **b**, illustrates the performance of the algorithm with and without the application of the metathesis heuristics (k in Fig. 3). (top row) Without the heuristics, the algorithm can identify the lowest-scoring isomorphous mapping by cutting only two single bonds—still, this solution is chemically incorrect. (bottom row) With the heuristics applied, the algorithm is not heavily (score + 4) penalized for cutting as many as four bonds (two σ and two π)—application of the heuristics costs much less (+0.5) and the algorithm can find the chemically correct isomorphous mapping

without full stoichiometry. Our algorithm outperforms the competition most decisively in mapping the 116 complex reactions—here, it mapped correctly 85.3% reactions compared to 44.8% for ReactionMap and 59.4% for MarvinJS.

Next, we performed similar comparisons on a set of 401 reactions that were not considered during training (for reaction miniatures with mappings, see Supplementary Note 3). This “test” set comprised: 100 relatively simple reactions with full stoichiometry taken from total syntheses published in *Org. Lett.*, *J. Am. Chem. Soc.*, and *J. Org. Chem.*; 100 typical reactions without full stoichiometry taken from patents; and 201 mechanistically complex reactions (92 stoichiometrically balanced and 109 unbalanced reactions) which include rearrangements and multicomponent reactions taken from recent literature (in most cases, after 2010 and from *Org. Lett.*, *J. Am. Chem. Soc.*, and *J. Org. Chem.*). For all 401 reactions, we compared the performance of our algorithm not only against MarvinJS and ReactionMap but also ChemDraw Prime (version 16.0.0.82) and Indigo (version 1.3.0 beta). The results summarized in Fig. 5 evidence that for simple reactions with full stoichiometry (green bars), all algorithms with exception of Indigo (65% correctness) are performing well—ours has 98% correctness, MarvinJS 85%, ReactionMap 95%, and ChemDraw 96%. For 100 reactions without full stoichiometry (blue bars), our algorithm is slightly more accurate than ChemDraw, MarvinJS, and

Indigo (100% vs 97%, 96% and 90% correctness, respectively) and significantly better than ReactionMap (11% correctness), which does not handle missing stoichiometry. As in the training set, the major differences are observed for complex reactions (red bars) for which our algorithm is correct in 84% of cases compared to 66% for ChemDraw, 62% for MarvinJS, 38% for Indigo, and 33% for ReactionMap. We make two comments regarding these results. First, the figures of merit for ChemDraw and Indigo are overestimated, because a sizable fraction of the results these mappers produce come without full mappings (see Supplementary Fig. 5)—though the fragments missing atom assignments are chemically unique and can be unambiguously assigned by a human chemist (on the other hand, such results might not be treated as correct during, say, automatic reaction rule extraction). If a more stringent criterion is applied that all atoms in the molecule must have unique numbers, the statistics for the two mappers are much worse (solid parts of the bars in the figure): for ChemDraw, 27% correctness on simple reactions with full stoichiometry, 16% for typical reactions without full stoichiometry, and 27% for complex reactions; for Indigo, 37% for simple reactions with full stoichiometry, 12% for typical reactions without full stoichiometry, and 16% for complex reactions. Second, it is instructive to compare the performance of all mappers on complex reactions with full stoichiometry (73 such reactions in the training set and 92 in the test set). For such reactions,

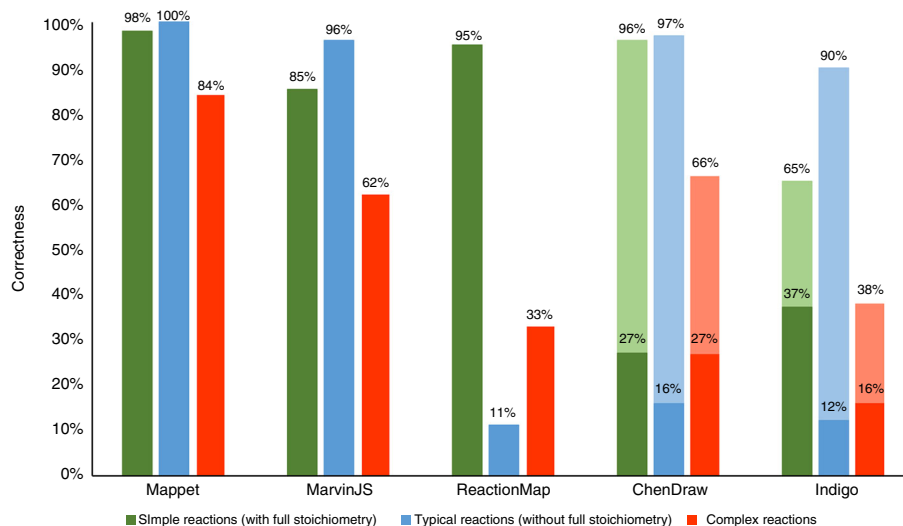


Fig. 5 Performance of various mappers quantified on the 401 reactions from the main test set. For ChemDraw and Indigo, solid parts of the bars correspond to the more stringent criterion that all atoms in the molecule must be assigned numbers. Shaded parts of the bars give the percentages of correct answers assuming a more lenient criterion allowing for unmapped but chemically unique atoms (see examples in Supplementary Fig. 6). ReactionMap does significantly better (~71%) on complex reactions with full stoichiometry (see main text)

correctness of ChemDraw, MarvinJS, and Indigo does not change much (respectively, 67%, 59%, 34%) but the performance of ReactionMap improves quite significantly (~71%)—though it is still perceptibly below our mapper (86% correctness for the stoichiometrically-balanced reactions in the training set and 87% for balanced reactions in the test set).

Some of these complex reactions we tested on, involving multiple mechanistic steps and/or rearrangements, are shown in Fig. 6. For example, reaction in Fig. 6a commences with a [3,3]-sigmatropic allylic azide rearrangement with subsequent intramolecular Schmidt-Aubé reaction to form bicyclic amide⁴¹. In Fig. 6b, cyclopropenyl lithium reagent generated in situ reacts with an aldehyde, followed by an intramolecular Diels-Alder reaction with a furan ring to form bridged bicyclic scaffold⁴². In Fig. 6c, a Lewis acid catalyzed semipinacol rearrangement creates bicyclo[3.2.1]octan-8-one scaffold⁴³. In Fig. 6d, a Lewis acid catalyzed oxa-Piancatelli rearrangement of an alcohol creates oxaspirocyclic scaffold⁴⁴. A multistep sequence in Fig. 6e involves cross-coupling of 2-bromofuran with an amide, intramolecular Diels-Alder reaction, thermal rearrangement of amidofuran to dihydro-2*H*-carbazolone and, finally, cyclization with allylamine to form the main scaffold of minfiensine alkaloid⁴⁵. On the flipside of the coin, the 16% of incorrectly mapped reactions are usually ones in which key, mechanistically important substrates or by-products are missing, those that comprise sequences of mechanistically complex rearrangements and unusual migrations of functional groups (changing atomic environments in non-trivial ways, cf. Supplementary Fig. 8), or those that are cascades of not necessarily complex but just too many (>4, 5) reactions⁴⁶ that could/should be written as separate transformations (Fig. 6f). As narrated earlier in the text, such corner cases cannot be overcome by simply adding more specialized heuristics since their application creates additional decoy solutions—especially when the heuristics' atom cores overlap—having comparable scores but ultimately yielding wrong atom assignments.

As the second test set, we considered the performance of our algorithm in mapping patent reactions from which reactivity scores for atom pairs¹¹ or reaction templates/cores⁴⁷ were previously derived and then used for, respectively, reaction prediction or retrosynthetic planning. Proper atom assignments in this and similar reaction sets are important since machine-extraction

of mapped reaction cores is common to most retrosynthetic design programs (Wiley/CAS ChemPlanner¹⁷, InfoChem's IC Synth¹⁶, BenevolentAI/Waller's¹², and MIT/Coley's⁴⁷ programs), though not of our Chematica platform^{13,40}. Here, we considered a subset of 50,000 reactions selected by Landrum and co-workers^{5,48} from the United States Patent and Trademark Office (USPTO) database to represent reactions most essential for medicinal chemistry (this collection was later used by the MIT team in the abovementioned studies¹¹ and⁴⁷). After further cleaning for chemically nonsensical entries (see Supplementary Fig. 9), we categorized the reactions according to the number of bonds that were altered (from one to six) and selected samples from each class at random to ultimately collect 281 examples (cf. Fig. 7 and Supplementary Note 4)—we note that this method of categorization and selection placed emphasis on the more complicated reactions that would otherwise be infrequent (~72% of the USPTO set alters one or two bonds) but are important in the context of learning non-trivial chemical reaction rules. When the reactions were mapped by human experts, we compared the correctness of mappings provided in the USPTO set (red bars in Fig. 7a) vs. the mappings generated by our algorithm (green bars). As seen, for reactions disconnecting/creating one to two bonds, there are no major performance differences; on the other hand, for more complex reactions changing more than two bonds, the percentage of correctly mapped reactions in the USPTO collection drops rapidly to 16–52% while it remains between 82 and 92% for our mapper. A manifestation of this trend are illustrated in Fig. 7b which shows two useful reactions, Diels-Alder cycloaddition and *N*-alkylation of amides, incorrectly mapped in the USPTO set. We observe that when the authors of ref. ¹¹ used such mappings to derive reactivity scores for atom pairs and then—using a deep neural network based on Weisfeiler-Lehman architecture—predicted outcomes of new reactions, they claimed that “the overall model performance does not depend strongly on atom mapping quality”. While this might be the case for some very simple chemistries, we have verified that if incorrect mappings are used to train the neural net, predictions of products of test reactions are, in vast majority, chemically nonsensical (see Supplementary Notes 5.2 vs. 5.3 and more thorough discussion in Section 8 of the Supplementary Information to ref. ⁴⁹). In a wider context, such examples help us understand why

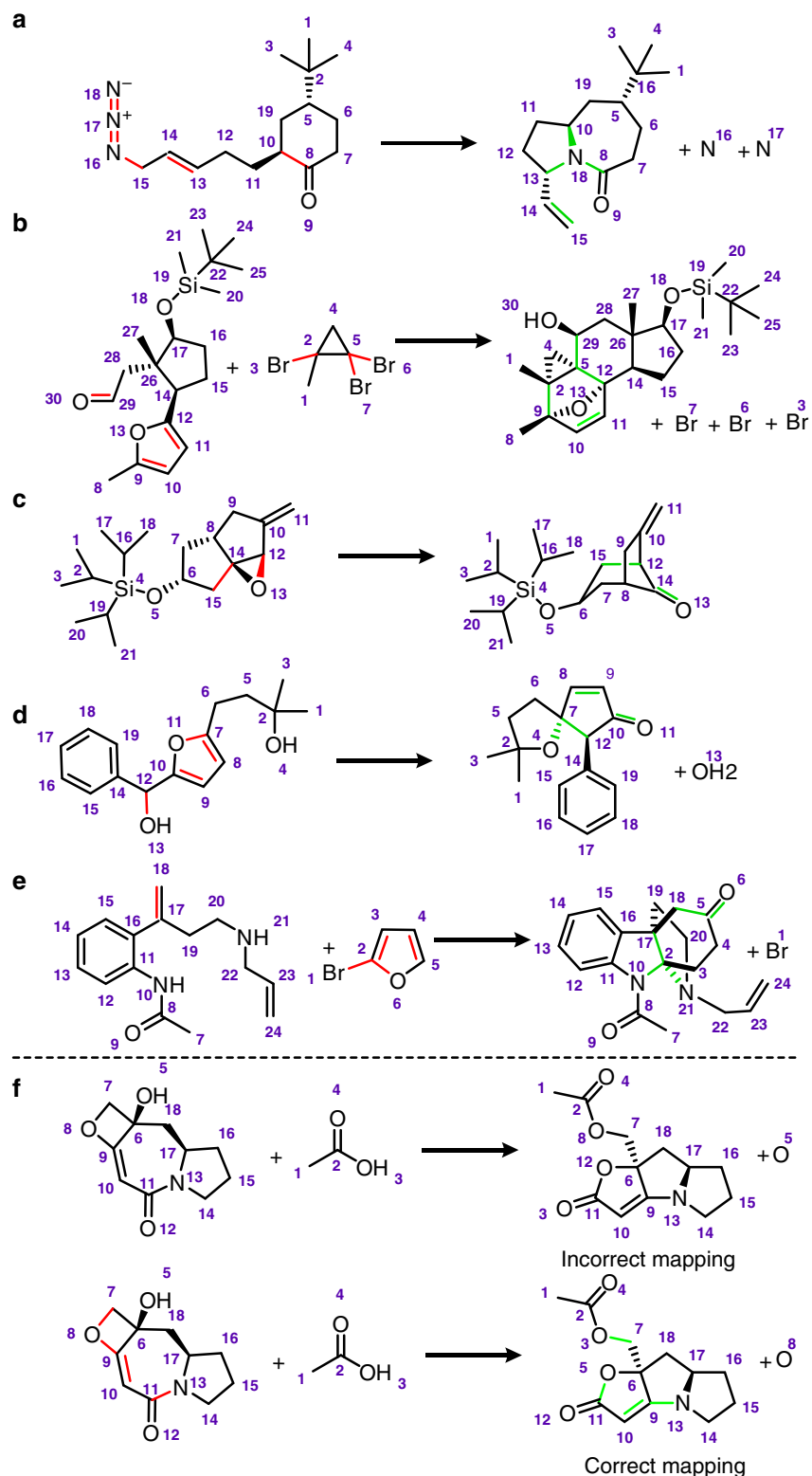


Fig. 6 Examples of mapped, complex reactions. **a–e** Reactions mapped correctly by our algorithm. **f** An example of a reaction in which the mapping found is incorrect. Correct mapping is shown in the bottom row. Bonds that are disconnected are colored in red whereas bonds created are marked green. For discussion, see main text and refs. 41–46 for chemical details

synthesis-planning softwares based on machine-extracted rules or reactivity indices that come with faulty atom mappings are generally not applicable to complex targets whose syntheses require the use of more advanced chemistries. We note, however, that other approaches are also being developed that do not require

atom mappings but, instead, base reaction predictions on reaction fingerprints¹⁴ or the so-called sequence to sequence models^{50,51}.

We sought additional validation from synthetic chemists outside of our group (three M.Sc. students, two Ph.D. candidates, two postdoctoral fellows, all listed in the Acknowledgments

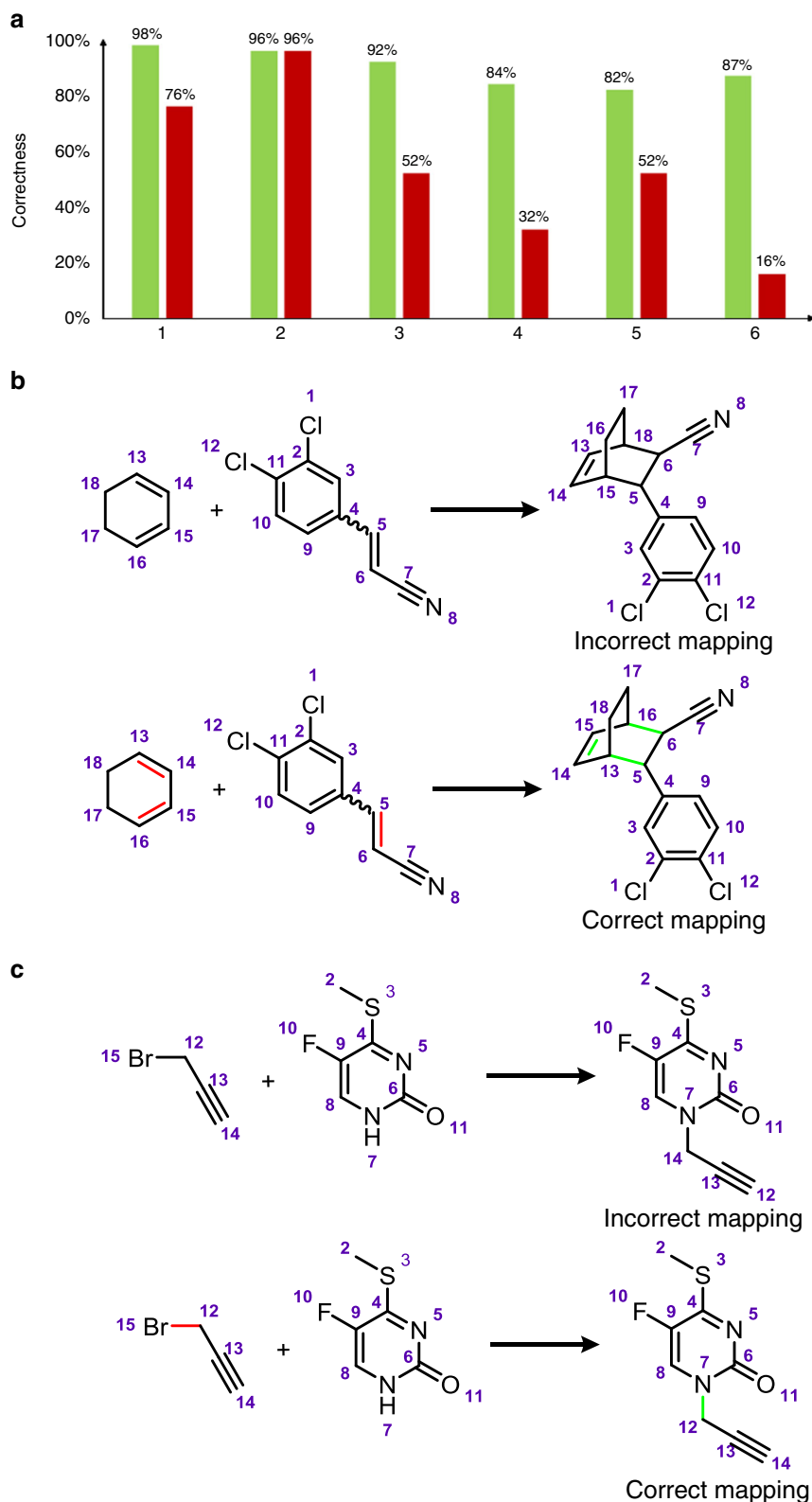


Fig. 7 Comparison of our algorithm against USPTO mappings. **a** Percentages of correct USPTO mappings (red bars) and those by our algorithm (green bars) categorized according to the number of bonds broken/created, from one to six. Statistics are based on 50 reactions each for one to five bonds being changed and 31 reactions for six bonds. We note that in addition to chemically meaningful reactions for which we compared the mappings, the USPTO set also contained a large fraction of nonsensical reactions that were likely due to human entry errors in the databases they used (e.g., missing key reaction partners, creating atoms “ex nihilo”, etc., see Supplementary Fig. 9). Such reactions as well as simple deprotections were not included in the statistics shown. **b, c** Examples of two reactions—Diels-Alder cycloaddition (**b**) and *N*-alkylation of amide (**c**)—mapped incorrectly in the USPTO set and correctly by our software. For more examples, see Supplementary Note 4

section). First, to benchmark our expert mappings, four of these chemists re-mapped a randomly-chosen sample of our 100 reactions—their mappings agreed with those of our internal experts. Then, to ensure that the reactions we chose for our tests were not in any way biased, all seven external chemists were asked to provide additional samples (25 reactions per person) they considered representative to modern synthetic chemistry. All these reactions—differing in the level of mechanistic complexity, provided to us in a typical synthetic notation (in 38.9% of cases, without full stoichiometry), and all listed in the Supplementary Section 5—were mapped by our algorithm and the results were compared against external mappings (in addition, our internal experts validated the external mappings once more). In the end, the algorithm provided 90% correct mappings for 100 reactions provided by Ph.D. candidates and postdocs, and 95% correct answers for reactions provided by M.Sc. students.

Finally, we considered algorithm's speed. Because our method uses several heuristics for the NP hard subgraph isomorphism problem, its speed is, in principle, exponential with respect to graph size—however, tests summarized in Supplementary Fig. 7 indicate that typical mapping times remain practical (91.5% of reactions mapped within 1 s, 96.5% in less than 10 s).

In summary, mapping of organic reactions is an example of a NP-hard problem for which prior attempts have been largely restricted to simple reaction types which chemists can typically map without much effort. In our approach, junction of graph theory and combinatorics with domain chemical knowledge (embodied in the minimal set of reaction heuristics) enables mapping of both simple and very complex organic reactions, including those that might challenge human experts. The graphical user interface of our algorithm is made freely available at <http://mapper.grzybowski-group.pl/marvinjs/> (see Supplementary Note 1.3 for a short tutorial) and we hope it will be useful for colleagues working on the applications we touched upon above (especially assignment of atoms in machine-extracted synthetic rules) and also in related fields, notably in the mapping of biochemical pathways (see examples in the Supplementary Note 1.4).

Data availability

The source code of the program is made available for academic users upon request to the corresponding authors.

Received: 12 August 2018 Accepted: 1 March 2019

Published online: 29 March 2019

References

- Grossman, R. *The Art of Writing Reasonable Organic Reaction Mechanisms* (Springer, New York, 2003).
- Clayden, J. *Organic Chemistry* (Oxford University Press, Oxford, 2001).
- Kraut, H. et al. Algorithm for reaction classification. *J. Chem. Inf. Model.* **53**, 2884–2895 (2013).
- Chen, L., Nourse, J. G., Christie, B. D., Leland, B. A. & Grier, D. L. Over 20 years of reaction access systems from MDL: a novel reaction substructure search algorithm. *J. Chem. Inf. Comp. Sci.* **42**, 1296–1310 (2002).
- Schneider, N., Stiefl, N. & Landrum, G. A. What's what: the (nearly) definitive guide to reaction role assignment. *J. Chem. Inf. Model.* **56**, 2336–2346 (2016).
- Rahman, S. A., Cuesta, S. M., Furnham, N., Holliday, G. L. & Thornton, J. M. EC-BLAST: a tool to automatically search and compare enzyme reactions. *Nat. Meth.* **11**, 171–174 (2014).
- Akutsu, T. Efficient extraction of mapping rules of atoms from enzymatic reaction data. *J. Comp. Biol.* **11**, 449–462 (2004).
- Heinonen, M., Lappalainen, S., Mielikainen, T. & Rousu, J. Computing atom mappings for biochemical reactions without subgraph isomorphism. *J. Comp. Biol.* **18**, 43–58 (2011).
- Latendresse, M., Malerich, J. P., Travers, M. & Karp, P. D. Accurate atom-mapping computation for biochemical reactions. *J. Chem. Inf. Model.* **52**, 2970–2982 (2012).
- Coley, C. W. et al. Prediction of organic reaction outcomes using machine learning. *ACS Cent. Sci.* **3**, 434–443 (2017).
- Jin, W., Coley, C. W., Barzilay, R. & Jaakkola, T. Predicting organic reaction outcomes with Weisfeiler-Lehman network. 31st Conference on Neural Information Processing Systems (NIPS), (Long Beach, CA, USA, 2017).
- Segler, M. H. S., Preuss, M. & Waller, M. P. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature* **555**, 604–610 (2018).
- Szymkuć, S. et al. Computer-assisted synthetic planning: The end of the beginning. *Angew. Chem. Int. Ed.* **55**, 5904–5937 (2016).
- Wei, J. N., Duvenaud, D. & Aspuru-Guzik, A. Neural networks for the prediction of organic chemistry reactions. *ACS Cent. Sci.* **2**, 725–732 (2016).
- Bøgevig, A. et al. Route design in the 21st century: the IC SYNTH software tool as an idea generator for synthesis prediction. *Org. Process Res. Dev.* **19**, 357–368 (2015).
- ICSYNTH: <https://www.nature.com/content/infochem/icsynth/index.html> (Accessed 16 Apr 2018).
- ChemPlanner: <https://www.cas.org/products/scifinder-n/chemplanner>, (Accessed 16 Apr 2018).
- Chen, W. L., Chen, D. Z. & Taylor, K. T. Automatic reaction mapping and reaction center detection. *Wiley Inter. Rev. Comput. Mol. Sci.* **3**, 560–593 (2013).
- Lynch, M. F. & Willett, P. The automatic detection of chemical reaction sites. *J. Chem. Inf. Comp. Sci.* **18**, 154–159 (1978).
- McGregor, J. J. & Willett, P. Use of a maximal common subgraph algorithm in the automatic identification of the ostensible bond changes occurring in chemical reactions. *J. Chem. Inf. Model.* **21**, 137–140 (1981).
- Funatsu, K., Endo, T., Kotera, N. & Sasaki, S. I. Automatic recognition of reaction site in organic chemical reactions. *Tetrahedron Comp. Meth.* **1**, 53–69 (1988).
- Körner, R. & Apostolakis, J. Automatic determination of reaction mappings and reaction center information. *J. Chem. Inf. Model.* **48**, 1181–1189 (2008).
- Crabtree, J. D. & Mehta, D. P. Automated reaction mapping. *J. Exp. Algorithm.* <https://doi.org/10.1145/1412228.1498697> (2009).
- First, E. L., Gounaris, C. E. & Floudas, C. A. Stereochemically consistent reaction mapping and identification of multiple reaction mechanisms through integer linear optimization. *J. Chem. Inf. Model.* **52**, 84–92 (2012).
- Fooshee, D. A. & Baldi, P. Reaction Map: an efficient atom-mapping algorithm for chemical reactions. *J. Chem. Inf. Model.* **53**, 2818–2819 (2013).
- Cook, S. A. The complexity of theorem-proving procedures. *Proc. Third Annu. ACM Symp. Theory Comput., STOC '71*, 151–158 (1971).
- Gonzalez, G. A. P. et al. Comparative evaluation of atom mapping algorithms for balanced metabolic reactions: application to Recon 3D. *J. Cheminform.* <https://doi.org/10.1186/s13321-017-0223-1> (2017).
- Mooock, T. E., Nourse, J. G., Grier, D. & Hounshell, W. D. Chemical structures Ch. *The implementation of atom-atom mapping and related features in the reaction access system (REACCS)* (Springer, Berlin, Germany, 1988).
- Morgan, H. L. The generation of a unique machine description for chemical structures—a technique developed at chemical abstracts service. *J. Chem. Doc.* **5**, 107–113 (1965).
- Clemens, J., Gasteiger, J. & Ugi, I. The principle of minimum chemical distance (PMCD). *Angew. Chem. Int. Ed.* **19**, 495–505 (1980).
- SPRESIweb: www.spresi.com (Accessed 20 July 2017).
- ReactionMapWeb: <http://cdb.ics.uci.edu/cgi-bin/reactionmap/ReactionMapWeb.py> (Accessed 20 July 2017).
- Marvin J. S., version 16.4.18; ChemAxon Ltd.: www.chemaxon.com (Accessed 20 July 2017).
- Cormen, T. H., Leiserson, C. E., Rivest, R. L. & Stein, C. *Introduction to Algorithms* Vol. 6. (MIT Press, Cambridge, 2001).
- Quinlan, J. R. & Michalski, R. S. *Machine Learning: An Artificial Intelligence Approach* (Springer Science & Business Media, 2013).
- Cordella, L. P., Foggia, P., Sansone, C. & Vento, M. A (sub)graph isomorphism algorithm for matching large graphs. *Ieee. Trans. Pattern Anal. Mach. Intell.* **26**, 1367–1372 (2004).
- Organic Syntheses: <http://www.orgsyn.org/> (Accessed 16 Apr 2018).
- Reaxys: www.reaxys.com (Accessed 16 Apr 2018).
- Kurti, L. & Czakó, B. *Strategic Applications of Named Reactions in Organic Synthesis* (Elsevier, Amsterdam, Netherlands, 2005).
- Klucznik, T. et al. Efficient syntheses of diverse, medicinally relevant targets planned by computer and executed in the laboratory. *Chem* **4**, 522–532 (2018).
- Liu, R., Gutierrez, O., Tantillo, J. D. & Aubé, J. Stereocontrol in a combined allylic azide rearrangement and intramolecular Schmidt reaction. *J. Am. Chem. Soc.* **134**, 6528–6531 (2012).
- Magnus, P. & Littich, R. Intramolecular cyclopropene-furan [2 + 4] cycloaddition followed by a cyclopropylcarbinyl rearrangement to Synthesize the BCD Rings of coristatin A. *Org. Lett.* **11**, 3938–3941 (2009).
- Plummer, Ch. W., Soheili, A. & Leighton, J. L. A tandem cross-metathesis/semipinacol rearrangement reaction. *Org. Lett.* **14**, 2462–2464 (2012).

44. Palmer, L. I. & de Alaniz, J. R. Rapid and stereoselective synthesis of spirocyclic ethers via the intramolecular Piancatelli rearrangement. *Org. Lett.* **15**, 476–479 (2013).
45. Li, G. & Padwa, A. Intramolecular Diels-Alder cycloaddition/rearrangement cascade of an amidofuran derivative for the synthesis of (\pm)-minfiensine. *Org. Lett.* **13**, 3767–3769 (2011).
46. Hickford, P. J. et al. Acid-catalyzed rearrangement of fused alkylideneoxetanols. *Org. Lett.* **9**, 4681–4684 (2007).
47. Coley, C. W., Rogers, L., Green, W. H. & Jensen, K. F. Computer-assisted retrosynthesis based on molecular similarity. *ACS Cent. Sci.* **3**, 1237–1245 (2017).
48. Schneider, N., Lowe, D. M., Sayle, R. A. & Landrum, G. A. Development of a novel fingerprint for chemical reactions and its application to large-scale reaction classification and similarity. *J. Chem. Inf. Model.* **55**, 39–53 (2015).
49. Beker, W., Gajewska, E. P., Badowski, T. & Grzybowski, B. A. Prediction of major regio-, site-, and diastereoisomers in Diels-Alder reactions by using machine-learning: the importance of physically meaningful descriptors. *Angew. Chem. Int. Ed.* **58**, 4515–4519 (2019).
50. Schwaller, P., Gaudin, T., Lanyi, D., Bekas, C. & Laino, T. “Found in Translation”: predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. *Chem. Sci.* **9**, 6091–6098 (2018).
51. Liu, B. et al. Retrosynthetic reaction prediction using neural sequence-to-sequence models. *ACS Cent. Sci.* **3**, 1103–1113 (2017).
52. Hopkins, M. H. & Overman, L. E. Stereocontrolled preparation of tetrahydrofurans by acid-catalyzed rearrangement of allylic acetals. *J. Am. Chem. Soc.* **109**, 4748–4749 (1987).

Acknowledgements

We gratefully acknowledge support from the Symfonia Award (UMO-2014/12/W/ST5/00592) from the Polish National Science Center (NCN). B.A.G. also gratefully acknowledges personal support from the Institute for Basic Science Korea, Project Code IBS-R020-D1. We thank the following colleagues for their help in selecting and mapping additional synthetic examples: Anna Domzalska and Dr. Michał Pieczykolan (both from the Institute of Organic Chemistry, Polish Academy of Sciences; Warsaw, Poland), Patryk Kasza (Jagiellonian University Medical College; Cracow, Poland), Dr. Dorota Jakubczyk (University of Lorraine, Laboratory of Molecular Engineering and Articular Pathophysiology; Vandoeuvre-Les-Nancy, France), and Geonhui Park, Jooyoung Oh and Hoyoung Jung (all from Ulsan Institute of Science and Technology, Department of

Chemistry; Ulsan, South Korea). We also thank Dr. Rafał Roszak and Dr. Wiktor Beker for their help in the statistical analyses of data and useful discussions.

Author contributions

W.J., S.S., and B.M.-K. co-developed and validated the algorithm. W.J. coded the algorithm. K.P., M.K., and J.R. developed the webservice. T.K. participated in chemical validations of the mappings. A.G. and B.A.G. conceived and supervised research. B.A.G. wrote the paper with contributions from S.S., B.M.-K., and A.G.

Additional information

Supplementary Information accompanies this paper at <https://doi.org/10.1038/s41467-019-09440-2>.

Competing interests: The authors declare no competing interests.

Reprints and permission information is available online at <http://npg.nature.com/reprintsandpermissions/>

Journal peer review information *Nature Communications* thanks Alexander Tropsha and the other anonymous reviewers for their contribution to the peer review of this work.

Publisher's note: Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2019