Bioinformatics, 33(24), 2017, 3955–3963
doi: 10.1093/bioinformatics/btx481
Advance Access Publication Date: 27 July 2017
Original Paper

Systems biology

Predicting novel metabolic pathways through subgraph mining

Aravind Sankar[1,†], Sayan Ranu[1,2,*,‡] and Karthik Raman[2,3,*]

[1]Department of Computer Science and Engineering, [2]Initiative for Biological Systems Engineering (IBSE), Interdisciplinary Laboratory for Data Sciences and [3]Department of Biotechnology, Bhupat and Jyoti Mehta School of Biosciences, Indian Institute of Technology (IIT) Madras, Chennai 600 036, Tamil Nadu, India

*To whom correspondence should be addressed.
[†]Present address: Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, IL, USA
[‡]Present address: Department of Computer Science and Engineering, IIT Delhi, New Delhi 110 016, India

Associate Editor: Oliver Stegle

Received on April 20, 2017; revised on July 17, 2017; editorial decision on July 25, 2017; accepted on July 26, 2017

Abstract

Motivation: The ability to predict pathways for biosynthesis of metabolites is very important in metabolic engineering. It is possible to mine the repertoire of biochemical transformations from reaction databases, and apply the knowledge to predict reactions to synthesize new molecules. However, this usually involves a careful understanding of the mechanism and the knowledge of the exact bonds being created and broken. There is a need for a method to rapidly predict reactions for synthesizing new molecules, which relies only on the structures of the molecules, without demanding additional information such as thermodynamics or hand-curated reactant mapping, which are often hard to obtain accurately.

Results: We here describe a robust method based on subgraph mining, to predict a series of biochemical transformations, which can convert between two (even previously unseen) molecules. We first describe a reliable method based on subgraph edit distance to map reactants and products, using only their chemical structures. Having mapped reactants and products, we identify the reaction centre and its neighbourhood, the reaction signature, and store this in a reaction rule network. This novel representation enables us to rapidly predict pathways, even between previously unseen molecules. We demonstrate this ability by predicting pathways to molecules not present in the KEGG database. We also propose a heuristic that predominantly recovers natural biosynthetic pathways from amongst hundreds of possible alternatives, through a directed search of the reaction rule network, enabling us to provide a reliable ranking of the different pathways. Our approach scales well, even to databases with >100 000 reactions.

Availability and implementation: A Java-based implementation of our algorithms is available at https://github.com/RamanLab/ReactionMiner.

Contact: sayanranu@cse.iitd.ac.in or kraman@iitm.ac.in

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Metabolic networks have been curated for hundreds of organisms in popular databases such as the Kyoto Encyclopedia of Genes and Genomes (KEGG; Kanehisa *et al.*, 2016), MetaCyc (Caspi *et al.*, 2012) and MetaNetX (Ganter *et al.*, 2013). These curated biochemical reaction databases represent the repertoire of biochemical conversions that known enzymes can catalyze. Enzymes, while being remarkably specific, also demonstrate the ability to convert a family of related substrates (e.g. alcohols), to a family of related products (e.g. aldehydes). An important challenge in metabolic engineering is the biosynthesis of novel molecules through heterologous expression of enzymes from other organisms. The ability to perform this *retrosynthesis* of novel molecules hinges on our ability to understand and generalize the abilities of the enzymes, in terms of the chemical reactions that they can catalyze and the substrates that they can act on.

Further, a deeper understanding of the biochemical transformations happening in metabolic networks can shed light on various fundamental questions in biology. For example, are there alternate ways to synthesize common central metabolites such as pyruvate? Why do cells prefer a particular pathway for the conversion of a metabolite such as glucose, to say, pyruvate (glycolysis)? There are also many knowledge gaps in our understanding of microbial metabolism; for example, there are a number of compounds known to be present in microbes, but the exact sequence of reactions and intermediates involved in their biosynthesis remain unknown. It is possible to bridge these knowledge gaps through a careful analysis of the metabolic networks, as we describe herein.

Since the seminal work of Corey and Wipke (1969), a number of algorithms have been developed to analyse (bio)chemical reaction networks, to predict pathways and novel routes for metabolite synthesis (Carbonell *et al.*, 2011, 2012, 2014a,b; Chen and Baldi, 2009; Hadadi *et al.*, 2016; Hatzimanikatis *et al.*, 2005; Kayala and Baldi, 2012; Latendresse *et al.*, 2014; Mithani *et al.*, 2009; Moriya *et al.*, 2010; Rahman *et al.*, 2009, 2014; Sivakumar *et al.*, 2016). For reviews, see (Hadadi and Hatzimanikatis, 2015; Medema *et al.*, 2012). Despite the availability of a wide array of reaction prediction methods, nearly all of them rely on the existence of query molecules in the reaction knowledge-base ('known' molecules in training data). A notable exception is PathPred (Moriya *et al.*, 2010), which can make predictions on unseen molecules. ReactionPredictor (Chen and Baldi, 2009; Kayala and Baldi, 2012) can also predict reactions for unknown molecules, but it is limited to specific classes of organic reactions, from which manually composed reaction transformation rules have been derived.

In this work, we present a general and fully automated method for predicting reactions between unknown (previously unseen) molecules. We do so by automatically learning biochemical transformation rules involving substructures of molecules from the reaction knowledge-base and searching for matching substructures in the unseen query molecule, both via subgraph mining techniques. The result is a scalable method that can be efficiently applied to predict novel metabolic routes in thousands of organisms. Notably, compared to previous methods, which use manually curated KEGG RPAIR mappings (Moriya *et al.*, 2010), or manually composed reaction transformation rules (Kayala and Baldi, 2012), we use no more information than a given metabolic reaction database and the chemical structures of the participating molecules. We also demonstrate two important applications of our method: first, we show how our method can be used to identify/recover biochemically preferred pathways between metabolites. Second, we show how pathways to known and novel/unseen compounds can be rapidly predicted. Our approach is very efficient, completely automated, scalable and performs with a higher degree of accuracy compared to state-of-the-art methods.

1.1 Related work

We now discuss how most previous approaches meet only a subset of the challenges mentioned above. The earliest work (Mavrovouniotis *et al.*, 1990) focuses on using stoichiometric constraints to identify feasible pathways, where reactions are classified as either being allowed, required or excluded from the pathways. Rosselló and Valiente (2004) proposed a chemical graph transformation approach to study metabolic networks. Rahnuma (Mithani *et al.*, 2009) employs a hypergraph model to represent a network between molecules for the prediction and analysis of pathways. An edge connecting two molecules denotes that it is possible to convert one to the other. Metabolic Tinker (McClymont and Soyer, 2013) is an open source web-server that uses the entire Rhea database to rank possible paths, based on thermodynamics. All the above techniques, however, fail to generalize for unknown query molecules. PathPred (Moriya *et al.*, 2010) uses a limited number of (Reactant, Product) pairs to predict pathways for a small subset of molecules. However, these pairs and their structural transformations are hand-curated and consequently, the technique is limited to a small collection of reactions. In our technique, we automatically learn both the pairing and the structural transformations.

EC-BLAST (Rahman *et al.*, 2014) proposes an algorithm to automatically search and compare enzyme reactions. Though their approach characterizes reactions using patterns derived from atom–atom mappings, they use additional chemical knowledge such as bond energies and do not address our precise problem of predicting chemical reactions. Furthermore, information on bond energies is not readily available. Kotera and co-workers (Kotera *et al.*, 2013) developed a method to learn enzymatic reaction likeness from metabolic reaction databases using chemical fingerprints. From Metabolite to Metabolite (FMM; Chou *et al.*, 2009) is a tool for predicting pathways based on the KEGG. Kotera *et al.* (2014) propose a supervised approach to predict multistep reaction sequences using step-specific classifiers. However, their model is limited to the reaction filling framework that restricts intermediate compounds to the training database, while we do not make any such assumption. Further, their prediction model lacks interpretability and may not be able to capture the chemically important characteristics of a reaction since it uses descriptor-based feature vectors to represent compound pairs. In a subsequent work, Yamanishi *et al.* (2015) propose a graph alignment based algorithm for pathway reconstruction using regioisomer-sensitive graph matching. In this algorithm, vertices of two graphs are aligned and then based on the alignment, a feature vector is constructed. These feature vectors are fed to a classifier to learn a classification model. In contrast, in our technique, the entire prediction is performed on the graph space, by identifying the reaction signatures (subgraphs), whose presence drives a particular reaction type. RouteSearch (Latendresse *et al.*, 2014) is a recent method to predict pathways using the MetaCyc database. This technique uses atom–atom mappings to search a metabolic network obtained from MetaCyc (Caspi *et al.*, 2012). Another very recent tool is Metabolic Route Explorer (MRE; Kuwahara *et al.*, 2016), which can rapidly predict pathways in several organisms and rank the pathways via a nice web interface. However, none of FMM, RouteSearch or MRE can predict on unseen molecules.

2 Methods

Figure 1 presents the pipeline of our reaction prediction algorithm. We represent each molecule as a graph, where atoms correspond to vertices and bonds correspond to edges. Given a database of metabolic reactions, we use an effective mapping method based on subgraph edit distance (He and Singh, 2006) to accurately map transformed metabolites in a reaction. Through graph mining, we then identify the specific subgraph within a graph (molecule) that is critical for a reaction to occur. We call these subgraphs the *reaction signatures*. For example, consider an alcohol to aldehyde conversion (see Fig. 2a), where RCH_2OH is converted to RCHO, by the enzyme alcohol dehydrogenase. We consider the subgraph corresponding to CH_2OH as the reaction signature, since the rest of the molecule remains unaffected. We then analyse the reaction signatures and characterize the changes they undergo during a reaction and summarize them as *reaction rules*. Connecting back to our example, the reaction rule in this case is CH_2OH changing to CHO. All reaction rules that are learned from the database are next consolidated in the form of a *reaction rule network* (RRN). In the RRN, each node is a reaction rule and two rules are connected by an edge if they can potentially form a reaction pathway. This completes the offline phase. In the online phase, given a query to find a pathway from molecule A to B, we analyse the structures of both A and B based on the reaction signatures they contain. From this analysis, A is mapped to a set of source nodes, and B is mapped to a set of destination nodes, in the RRN. Consequently, the prediction problem reduces to finding (optimal) paths between the source and destination nodes in the RRN.

2.1 Problem formulation

In this section, we formulate our prediction problem and define the concepts and notations central to our work. We represent each molecule as an undirected graph. A graph $g(V, E)$ is composed of a set of vertices $V = \{v_1, \ldots, v_n\}$ and a set of edges $E = \{e = (v_i, v_j) \mid v_i, v_j \in V\}$. Each vertex and edge have labels denoted $l(v)$ and $l(e)$ respectively. The *size* of a graph is $|E|$. Figure 2(b) shows the graph representation of a molecule. Atoms correspond to vertices, bonds correspond to edges and bond orders correspond to edge labels.

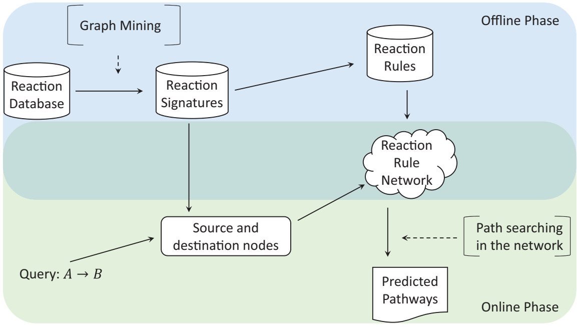


Fig. 1. Pipeline of the reaction prediction algorithm. The figure outlines both the offline and online phases of the algorithm. The offline phase involves graph mining of the reaction database to identify reaction signatures, from which reaction rules are subsequently identified and embedded in a reaction rule network (RRN). In the online phase, we search the RRN and predict suitable pathways, on the arrival of a query $A \rightarrow B$

The input to our problem is a dataset of chemical reactions, R. Specifically, we use the KEGG database, as detailed in Supplementary Methods Section 1.5. A reaction $\mathcal{R}$ contains two sets of graphs (or molecules): the first set contains the *reactants* and the second set contains the *products* synthesized. We use $RS(\mathcal{R})$ to denote the reactant set in $\mathcal{R}$ and $PS(\mathcal{R})$ to denote the products. A *pathway* $P(A, B)$ from a molecule A to B is a chain of reactions $\mathcal{R}_1, \ldots, \mathcal{R}_n$ such that $A \in RS(\mathcal{R}_1), B \in PS(\mathcal{R}_n)$, and $PS(\mathcal{R}_i) \cap RS(\mathcal{R}_{i+1}) \neq \phi \, \forall 1 \leq i \leq n-1$, i.e. there is at least one metabolite shared between the product set of one reaction and the reactant set of the next.

We now define the pathway prediction problem as follows: Given a training database of reactions (and the structures of the constituent molecules), learn a prediction model $\mathcal{M}$. $\mathcal{M}$ should support the prediction query $Q(\mathbb{S}, T)$, where $\mathbb{S}$ is a (set of) source molecule(s) and T is the target molecule. Given this query, $\mathcal{M}$ should produce a pathway $P(A, T)$ where $A \in \mathbb{S}$.

An important aspect of our formulation is that we do not make any assumption of the source or the target molecules being present in the reaction database. The only information we use to learn the prediction model are the structures of the molecules, which is easily available.

2.2 Mining reaction patterns

Our goal in this section is two-fold. First, we identify the *reaction patterns* existing in the training database. Second, for any given molecule in the reactant set, we should be able to predict the patterns that are applicable on the reactant. To understand what a pattern is in our context, let us revisit Figure 2(a). We claim that both reactions follow the same pattern because: (i) in both the alcohol molecules, the exact same subgraph (highlighted in red) is affected, while the remaining portions remain unaltered, (ii) the affected subgraphs undergo an identical change and (iii) the oxidizing agent undergoes an identical change to form a water molecule.

In other words, if the same structural change happens in one or more reactions, then that is a pattern. To quantify the *structural change*, we first need to construct a mapping between the graphs in the reactant set to those in the product set. More specifically, the alcohol molecules should be mapped to the aldehyde molecules and the oxidizing agent should be mapped to water. The comparison in the structure of the mapped molecules allows us to quantify the change. We call this operation *reactant–product mapping (RPM)* and use the notation $RPM(A, B)$ to denote that a reactant A has been mapped to a product B of the reaction. Clearly, a wrong

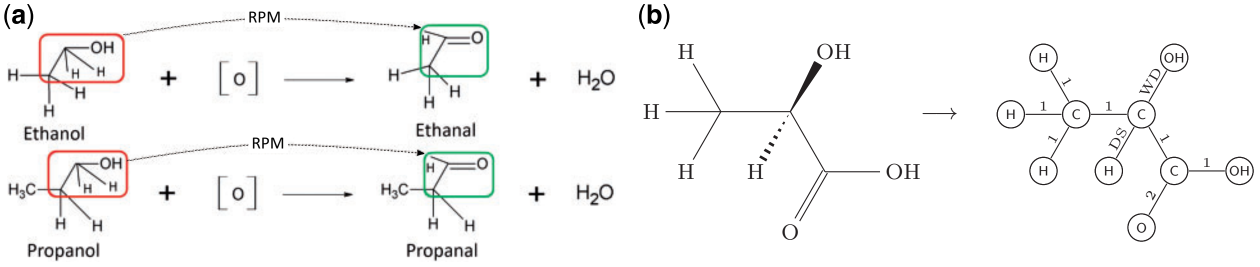
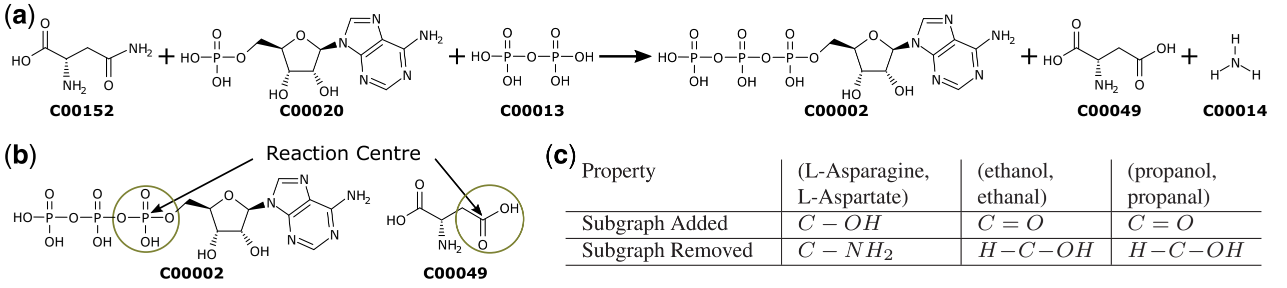


Fig. 2. A simple illustration to motivate our approach. (**a**) Conversion of ethanol and propanol (alcohols) to ethanal and propanal (aldehydes) respectively. Vertices without explicit labels represent Carbon atoms. Notice that although the reactions involve different molecules, the changes (highlighted in boxes) are identical. (**b**) Representing D-Lactic acid as a graph. Note that double bonds are indicated by a changed edge label, as are *wedges* and *dashes* that represent bond stereochemistry



Property	(L-Asparagine, L-Aspartate)	(ethanol, ethanal)	(propanol, propanal)
Subgraph Added	$C-OH$	$C=O$	$C=O$
Subgraph Removed	$C-NH_2$	$H-C-OH$	$H-C-OH$

Fig. 3. Illustration of reaction signature. (**a**) An example reaction, illustrating the conversion of the amino acid L-Asparagine (C00152) to L-Aspartate (C00049). The other reactants/co-factors in this reaction include ATP (C00002), AMP (C00020), Diphosphate (C00013) and Ammonia (C00014). (**b**) The reaction centres and signatures (colored circle) in the reaction in (a). (**c**) The structural changes in the reactant–product pairs, in terms of subgraphs added and removed (Color version of this figure is available at *Bioinformatics* online.)

mapping (such as mapping alcohol to water) would produce spurious results. RPM is essentially an automated approach to map reactants and products, similar to the RPAIR concept used in KEGG (Oh *et al.*, 2007), but we compute it only using the molecule structures and without resorting to any manual curation.

Clearly, computing the structural change is possible only after the RPM is constructed. To detect RPMs, we use *subgraph edit distance*, as we discuss in Supplementary Methods Section 1.3. We demonstrate the robustness of our RPM by comparing our matched reactant–product pairs with the manually curated KEGG RPAIR database (discussed in Supplementary Methods Section 2.2). To quantify the changes due to the reaction, we first identify the *reaction centres*. Subsequently, we identify the *reaction signatures*, or the motifs we consider necessary for a reaction to occur (see Supplementary Methods Section 2.2.1).

2.2.1 Reaction centres

The reaction centre for an RPM pair (A, B) is the set of vertices in the product B to which new edges are added, or existing edges removed, during its transformation from A. The reaction centre can easily be determined from the mapping ϕ corresponding to $sed(A, B)$. Specifically, it is a vertex v in the product B, such that $l(v) = l(\phi(v))$, but there exists an edge (v, v'), where $l(v') \neq l(\phi(v'))$. Recall, $l(v)$ denotes the label of v. For instance, consider the reaction in Figure 3(a), and particularly focus on the pair (C00049, C00152). In this pair, the conversion involves a removal of the OH group and addition of the NH_2 group. Thus, we have one reaction centre, which is the carbon atom attached to the bond involved with the change. The reaction centre is explicitly shown in Figure 3(b). Although it is more common to see one reaction centre in a pair, multiple reaction centres are possible.

2.2.2 Reaction signature

The reaction centre only tells us the location of change. It does not necessarily tell us the reason, or the conditions necessary, for the change to occur. To predict pathways, we need to identify the conditions required for a reaction to happen. We build our prediction model based on the hypothesis that two molecules would undergo a similar change in a reaction if they contain a common 'key' substructure that drives the forming or breaking of chemical bonds. Our hypothesis is motivated by the fact that many enzymes, such as alcohol dehydrogenases that convert alcohols to aldehydes, show a specificity towards the type of subgraph, *i.e.* sub-structure present in the reactants (Kauzmann, 1959; Palmer, 2007). Since the reaction centre is the location of the change, a straightforward approach would be to assign the reaction centre as this 'key' subgraph. However, a single atom (or vertex) does not capture all of the atom-level interactions that take place. For instance, consider the reaction centre in L-Asparagine (C00152; see Fig. 3a), which is a Carbon atom. Here, the Carbon is not only interacting with the NH_2 group that gets replaced with the OH group, but also with the adjacent Oxygen and Carbon atoms. The strength of the C=O and C–C bonds, their charges, geometries etc. all play a role in the breaking of the C–NH_2 bond and its eventual replacement with C–OH. To generalize, the direct neighbours of the reaction centre influence the reaction. Based on this intuition, we define a *reaction signature*, $S(V_S, E_S)$ as the immediate ('one-hop') neighbourhood of the reaction centre in the product of the (A, B) pair.

The reaction signatures of the two reactant–product pairs in the reaction in Figure 3(a) are shown in Figure 3(b). It is easy to see that the reaction signature is a subgraph of the product. Note that when there are multiple reaction centres, there are multiple reaction signatures as well, where each signature represents the neighbourhood around the corresponding reaction centre. In general, the reaction centres identify the locations of change, and the reaction signatures encode the potential driving factor behind the change. A formal description of how reaction signatures are computed and stored can be found in Supplementary Methods (Section 1.4).

2.2.3 Detecting the change in a mapped reactant–product pair

Conceptually, in a pair of mapped reactant and product molecules (A, B), we want to store $\Delta = B - A$, where Δ is the difference between the structures. Furthermore, given only B and Δ, we should be able to re-construct A. As we will see later, the ability to reconstruct the reactant A from just Δ and the product B lies at the core our of our algorithm's ability to predict on unseen molecules. The reaction signature can change through either the addition or removal of subgraphs, as detailed in Supplementary Methods Section 1.4.

To illustrate, Figure 3(c) shows the structural changes for three different pairs. The first pair is from Figure 3(a). The other two pairs correspond to the reactions in Figure 2(a). Notice that since both reactions in Figure 2(a) involve the conversion of alcohol to aldehyde, their structural changes (along with the reaction centres and signatures, which are not shown in Fig. 3c) are identical. The above illustration not only showcases how we capture structural changes in a reaction, but also demonstrates our precise ability to detect a common pattern among reactions. Armed with this technique, we next formulate the idea of a *reaction rule*.

2.3 Reaction rules

Given a database of reactions R, for each reaction $\mathcal{R}$, we identify all of its reactant–product pairs. From each pair (A, B), we extract and store the following information: (i) the reaction signature, (ii) the reaction centres, (iii) the subgraphs added and removed and (iv) all reactants in $\mathcal{R}$ except A. These reactants are the co-factors or *helper* reactants that facilitate the reaction. For example, the oxidizing agent would be stored as the helper reactant for (ethanol, ethanal) and (propanol, propanal) pairs in Figure 2(a). For the (C00152, C00049) pair in the reaction in Figure 3(a), both C00020 and C00013 would be stored.

We denote the above information, which is extracted from each (A, B) pair, as $L(A, B)$, the reaction rule. Note that we do not store the pair (A, B) itself; we only store the structural change and its associated information. $L(A, B) = L(C, D)$ if all of the four items listed above are identical. For example, $L(ethanol, ethanal) = L(propanol, propanal)$.

Given a support threshold θ, $L(A, B)$ is called a reaction rule if $L(A, B)$ occurs more than θ times in the database. Essentially, a reaction rule encodes the conditions required for a reaction to produce a predictable output. The support threshold controls the minimum number of times a pattern of structural change must be seen to be considered a reaction rule. The role of the support threshold is identical to its purpose in the well known problem of association rule mining. Since novel pathway identification between rare molecules is of critical importance, we err on the side of exploration, and set the default $\theta = 1$, which means any structural change is a pattern, even if it does not repeat across multiple reactions.

2.4 Pathway prediction

We now discuss how the reaction rules described above can be employed to predict synthesis of a target product. A reaction rule serves two purposes: firstly, given any target product molecule, detect whether the rule is applicable on the molecule. If the rule is applicable, we must predict the reactants required to synthesize the given product. In other words, the reaction rule is used to simulate the reaction in the reverse direction by predicting the reactant given the target product molecule. We introduce two graph operators: graph addition and subtraction, which enable the above. An example of the operations is shown in Supplementary Figure S1(a).

Supplementary Algorithm S1 presents the pseudocode of applying a reaction rule. Let B be a target product and L be the reaction rule that we want to apply on B if chemically feasible, i.e. if the rule is applicable, based on the presence of appropriate subgraphs. Recall our hypothesis that the presence of the reaction signature is the cause of the reaction. Second, due to the reaction, the 'Subgraph Added' of L gets attached at the reaction centre c. Thus, we first merge the reaction signature with the 'Subgraph Added' to create a single merged graph m. If m is a subgraph of B, then L is applicable on B. If the check passes, we proceed to the next step of formulating the reactants that can synthesize B. Since the reaction centre is present both in the signature and the 'Subgraph Added', m is guaranteed to be connected.

First, we construct the reactant pair of B using L. We remove the 'Subgraph Added' from B (line 3) and then merge the 'Subgraph Removed' component with B to create the reactant pair A (line 4). Finally, the helper reactants in L are fetched and their reaction with A is predicted to synthesize B (line 5). Note that neither B nor A is required to be present in the training database—only a matching subgraph need be present. An example is illustrated in the Supplementary Figure S1.

2.5 Reaction rule network

While we have described above, the procedure to predict a reaction that could synthesize a target molecule (also see Supplementary Algorithm S1), our goal is to predict pathways—essentially a chain of reactions. Furthermore, between a source and a target molecule, there could be hundreds of pathways. How do we identify and rank only the top-k best paths? To overcome these challenges, we propose the idea of an *RRN*.

Each node in the RRN corresponds to a reaction rule, and we want to ensure the following property: if there exists a pathway $P = \{R_1, \ldots, R_n\}$ from molecule A to B, such that reaction R_i happens through rule L_i, then, there should be a path from L_n to L_1 in the RRN. Towards that goal, we notice that rules L_1 and L_2 can be applied consecutively if the product of L_1 is a reactant in L_2. In such a case, we should have a directed edge from L_2 to L_1. However, neither the product nor the reactant may be present in the database. We need to capture this dependency between rules L_1 and L_2 only from the structural change information that we store.

To capture all of these properties, we use two graph operators—graph addition and subtraction (illustrated in Supplementary Fig. S1a) to formally define the *RRN* as follows:

Let $\mathbb{L}$ be the set of all rules mined from our training database. The RRN $N(V_N, E_N)$ is a directed graph where $V_N = \mathbb{L}$. Let $g_2 = L_2.signature - L_2.subgraphAdded + L_2.subgraphRemoved$ and $g_1 = L_1.signature + L_1.subgraphAdded$ and $e = (L_2, L_1) \in E_N$ if $g_1 \subseteq g_2$. We define an edge from L_2 to L_1 if rule L_1 is applicable on a reactant obtained by applying L_2 on any compound (Note that a reaction rule is applied in the reverse direction). g_2 (as defined above) is the subgraph that must be present on any reactant obtained by applying L_2 on a compound. On the other hand, g_1 is the subgraph that must be present on any product on which L_1 is applicable (follows from Supplementary Algorithm S1). Thus, if g_1 is a (subgraph isomorphic) subgraph of g_2, then rule L_1 can be applied on the reactant obtained by applying L_2 on any compound. In other words, if we visualize the process in terms of reactions, the product of L_1 can feed in as a reactant to L_2. To illustrate the RRN, consider a training database where in addition to the two reactions in Figure 2(a), we also have the oxidation of ethanal to ethanoic acid shown in Supplementary Figure S1(c). Furthermore, we consider every unique structural change as a pattern. Thus, there are two reaction rules; rule L_1 corresponding to the conversion of alcohol to aldehyde, and rule L_2 corresponding to the conversion of ethanal to ethanoic acid. The reaction signature, subgraph added and subgraph removed for L_2 is also shown in Supplementary Figure S1(c). The resultant g_2 and g_1, as shown in Supplementary Figure S1(d), are isomorphic, and consequently, there is an edge from L_2 to L_1 in the resultant RRN.

The formalization of the RRN completes the offline model building component. Next, we discuss the online query $(\mathbb{S}, T)$, where the goal is to find a pathway from A to T where $A \in \mathbb{S}$ is one of the source molecules.

2.6 Answering queries on the RRN

To illustrate our query answering strategy, we continue with the RRN outlined above. Suppose the query is to find a pathway from hexanol to hexanoic acid (Supplementary Fig. S1e). Note that neither of the query molecules are in the reaction database. We initiate by searching for a rule that is applicable on the target molecule, hexanoic acid. In our two-node network, rule L_2 is applicable (line 1 in Supplementary Algorithm S1). On applying L_2 on hexanoic acid, hexanal is generated as the reactant pair. Since hexanal is not the source molecule, we continue searching by applying the adjacent rule L_1. Since L_1 is connected from L_2, we are guaranteed that L_1 is applicable on the reactant produced by L_2, which is hexanal. On applying L_1 on hexanal, hexanol is generated as the reactant pair, which completes the query since it is the source molecule. The resultant pathway is therefore hexanol$\xrightarrow{L_1}$hexanal$\xrightarrow{L_2}$hexanoic acid.

To generalize the above strategy, we first identify nodes (or rules) that are applicable on the target molecule. From each of these rules, a reactant is generated. If the reactant is one of the source molecules then we stop. Otherwise, we continue exploring each possible path using breadth-first search (BFS) either till all paths are exhausted or a source molecule is reached. Exploration using *BFS* guarantees that the first pathway found is the shortest, in terms of length. The exploration algorithm can easily be generalized to find the *k* shortest paths as well. While *BFS* is simple, it is often not scalable in a large RRN due to the large number of paths that exists. Furthermore, the BFS strategy does not use the knowledge of the source molecule to optimize the searching process. To overcome these weaknesses, we explore an alternative algorithm, based on *best-first search* (Russell and Norvig, 2003).

2.6.1 Heuristic H_d: minimizing structural changes in every step

We hypothesize that nature avoids reactions that cause drastic alterations to the structure of the reactant. This can also be appreciated in terms of the enzymes—enzymes are highly specialized and perform an incremental structural change to a substrate, rather than wholesale structural changes. We model this effect through a distance function that minimizes the total structural change in a pathway, in addition to minimizing the distance to a source molecule *A*. Specifically, the optimization function at a specific pathway $P = \{X_1, \ldots, X_n\}$ of *n* molecules (*n* − 1 reactions) minimizes the function below:

$$H_d(P, \mathbb{S}) = \sum_{i=1}^{n-1} ged(X_i, X_{i+1}) + \min_{\forall A \in \mathbb{S}}\{ged(X_n, A)\} \qquad (1)$$

where $ged(g, g')$ is the *edit distance* between graphs g and g' (Zeng *et al.*, 2009). Edit distance between two graphs is defined analogously to subgraph edit distance. Specifically, it is the minimum number of edits required to convert g to g'. The primary difference with $sed(g, g')$ is that g is converted to g' instead of a subgraph of g'. Consequently, $ged(g, g')$ is symmetric. Based on $H_d(P, \mathbb{S})$, we optimize search paths using best-first search, as listed in Supplementary Algorithm S2.

3 Results

In this section, we establish that our pathway predictions are accurate, and that the proposed technique is scalable to large reaction databases. Ours is the first technique that is fully automated, can answer queries on unseen molecules, and requires no information other than the structure of the molecules. Due to this simplicity of our technique, we are the first to scale to a database as large as 150 000 reactions.

Our major results are three-fold. First, we query on those source and target molecules present in the training database. The presence of query molecules in the training set is enforced only to allow us to compare the performance with the state-of-the-art pathway prediction techniques such as *RouteSearch* (Latendresse *et al.*, 2014) and *MRE* (Kuwahara *et al.*, 2016). We demonstrate how our heuristic H_d picks up natural biosynthetic pathways very frequently, much more than other state-of-the-art methods. We argue that our heuristic is therefore a robust method to rank pathways based on biological plausibility. Second, we remove the constraint of requiring the source molecules in the training database and show that we predict viable retrosynthetic pathways for known and new molecules. Finally, we show that our results are accurate, by means of cross-validation, and that our algorithm can scale well for very large reaction databases.

3.1 H_d consistently picks up natural pathways with high probability

In any pathway prediction algorithm, all predicted pathways are ranked according to some score, and finally the top-*k* highest scoring paths are studied further for feasibility. Ranking the predicted pathways is very important since there are often hundreds of paths between two molecules, and a high rank should signify high biochemical plausibility. As discussed earlier, we use Equation (1) as the ranking function in our algorithm. To benchmark, we choose 20 pathways involved in the biosynthesis of amino acids and important precursors in central carbon metabolism, similar to those used in Carbonell *et al.* (2011) (see Supplementary Table S1). For the selected pathways, we predict by querying using their source and target molecules and extract the top 10 predicted paths. The training database for this experiment corresponds to the reaction set of *Escherichia coli*. Table 1 presents the rank of the actual pathway by each of the techniques. As clearly evident, the actual pathway consistently ranks among the top 10 in our algorithm, while being mostly absent in RouteSearch. MRE is able to predict only 10 of the 20 pathways. Although MRE occasionally ranks the correct result higher than our method, it clearly lags behind our method in the overall head-to-head comparison (4–14 with 2 ties). These results point towards the superior ability of our technique to identify pathways reliably, and also rank the biologically favoured pathways much higher. We have prioritized our comparisons against the recent methods, MRE and RouteSearch; other methods that focus on retrosynthesis, such as FMM and BNICE (Hadadi *et al.*, 2016) cannot be restricted to a single organism, while RetroPath and ReactPRED do not predict paths between pairs of compounds.

3.2 Retrosynthetic predictions compare favourably with other methods

In addition to the pathways we outlined above, we here show that we perform comparably or better than MRE, in nearly all retrosynthesis examples discussed in Kuwahara *et al.* (2016). We predict retrosynthesis pathways for commercially important metabolites, such as itaconate, naringenin, 1,3-propanediol, xylitol, etc. We find that in a majority of cases, we are able to recover known pathways or predict shorter biologically plausible pathways for retrosynthesis. We summarize our retrosynthesis predictions in Supplementary Table S2, alongside comparisons with MRE/FMM. We here note that there are many more methods, such as Pathway Hunter Tool (Rahman *et al.*, 2005), Metabolic Tinker (McClymont and Soyer, 2013), RetroPath (Carbonell *et al.*, 2014a), ReactPRED (Sivakumar *et al.*, 2016) and BNICE (Hadadi *et al.*, 2016; Hatzimanikatis *et al.*, 2005). We have prioritized our comparisons against MRE, the most recent method and refrained from repeating the observations of Kuwahara *et al.* on the performance of other techniques. It is also important to note that the main strength of our method is its ability to predict pathways between a pair of source–target metabolites and rank the putative pathways based on biological plausibility (Heuristic H_d), which aligns well with methods such as MRE/FMM. On the other hand, the key strength of methods such as RetroPath/

Table 1. Pathway Prediction comparison of our algorithm (specifically, using the heuristic H_d) versus RouteSearch and MRE

ID	Source	Target	Rank		
			H_d	RouteSearch	**MRE**[a]
1	α-D-Glucose (C00267)	D-Glyceraldehyde (C00118)	**1**	—	—
2	D-Glyceraldehyde (C00118)	Pyruvate (C00022)	**2**	—	13
3	5-Phospho-α-D-ribose (C00119)	L-Histidine (C00135)	**7**	—	—
4	Phosphoribulosyl-formimino-AICAR-phosphate (C04916)	L-Histidine (C00135)	**1**	—	—
5	D-Galacturonate (C00333)	Pyruvate (C00022)	2	—	**1**
6	D-Erythrose (C00279)	Pyridoxal phosphate (C00018)	**6**	—	119[b]
7	L-Threonine (C00188)	L-Isoleucine (C00407)	**1**	—	**1**
8	GTP (C00044)	7,8-dihydropteridine (C04874)	**4**	—	No path
9	7,8-Dihydroneopterin 3'-triphosphate (C04895)	Dihydrofolate (C00415)	3	—	**2**
10	L-Aspartate (C00049)	2,3,4,5-Tetrahydrodipicolinate (C03972)	**2**	—	—
11	L-Aspartate (C00049)	L-Threonine (C00188)	**3**	—	—
12	Oxaloacetate (C00036)	L-Glutamate (C00025)	**3**	—	—
13	β-D-Glucose (C01172)	D-Glyceraldehyde (C00118)	6	—	**1**
14	2-Oxobutanoate (C00109)	L-Isoleucine (C00407)	**1**	—	**1**
15	Chorismate (C00251)	L-Tryptophan (C00078)	—	**1**	**1**
16	Shikimate (C00493)	L-Tyrosine (C00082)	**1**	**1**	38
17	L-Glutamate (C00025)	L-Ornithine (C00077)	8	**1**	—
18	Phosphoenolpyruvate (C00074)	L-Aspartate (C00049)	**3**	—	—
19	Phosphoenolpyruvate (C00074)	L-Asparagine (C00152)	**3**	—	—
20	L-Glutamate (C00025)	L-Proline (C00148)	**3**	—	—

Note: The source and target molecules are indicated along with their KEGG CIDs. Bold-faced rank displays the winning algorithm for each row. The details of the 20 pathways can be found in Supplementary Table S1.

[a] — in this column indicates the pathway is not found, in the top 200.

[b] Skips a step.

ReactPRED is to predict and rank multiple pathways for the synthesis of a metabolite, without a fixed 'source' metabolite.

For itaconate, an important value-added precursor from biomass (Werpy and Petersen, 2004) we recovered the same path as predicted by FMM. For production of naringenin, an important plant secondary metabolite and resveratrol, we find the same pathway identified by MRE and FMM. For the production of xylitol, our top-ranked pathway is shorter than that proposed by MRE, and agrees with FMM. For artemisinic acid, an important anti-malarial drug, synthesized in metabolically engineered *S. cerevisiae* (Ro *et al.*, 2006), we were able to predict the same path as MRE, from HMG-CoA, although this differs from Ro *et al.* (2006). For paths from acetyl-CoA to artemisinic acid, and chorismate to L-Tryptophan, the top ranked paths from our algorithm are not very relevant, perhaps due to the occurrence of very high-degree metabolites, such as acetyl-CoA and pyruvate.

We also predicted pathways to three volatile organic compounds in *Mycobacterium tuberculosis* that are not present in the KEGG database. These pathways are hitherto unknown, but we have previously predicted the synthesis routes on the basis of enzyme biochemistry and sequence analyses (Bhatter *et al.*, 2017). The exact pathways predicted previously were ranked highest by our algorithm, for methyl nicotinate and methyl *p*-anisate. For methyl phenylacetate, the correct path was ranked third, superseded by two other pathways involving metabolites not found in *M. tuberculosis*. These results highlight the ability of our method to predict biologically plausible pathways to even synthesize previously unseen molecules. Other methods such as ATLAS, RetroPath, FMM and MRE do not have the target molecules in their database and are therefore unable to predict paths.

Furthermore, we also examined some of the pathways evolved by organisms to degrade anthropogenic chemicals such as pentachlorophenol (Cai and Xun, 2002; Copley, 2009). We find that we are able to generate the identical pathway between pentachlorophenol (C02575) and Maleylacetate (C02222), as indicated in Supplementary Table S2. It is interesting to note that this predicted pathway is one of several possible pathways, given that we can apply many reaction rules to every intermediate. We also find that MRE and FMM are unable to find any pathways between these compounds, illustrating the importance of our ability to generalize reaction rules, as well as handle novel molecules. MRE and our approach both correctly predict another pathway where atrazine (C06551) is converted to urea-1-carboxylate (C01010). Together, these results illustrate the ability of our approach to not only predict retrosynthetic pathways, but also possible pathways that organisms may use to metabolize xenobiotics. Importantly, our heuristic of minimizing the metabolic transformations in a reaction enables us to recover the very pathway these organisms have evolved to breakdown xenobiotics.

3.3 Cross-validation illustrates the high accuracy of our pathway predictions

Given that the size of most metabolic databases is only of the order of 10 000 reactions, we synthetically expanded the KEGG database to >150 000 reactions as detailed in the Supplementary Methods Section 1.6. First, we evaluate through 5-fold cross-validation. Specifically, we split the KEGG Dataset into five parts, learn the training model on four parts and predict on the fifth part. This process is repeated to cover each part as the test set. For our prediction query, we pick arbitrary pathways from the test set and check if the exact pathways are predicted. We always ensure that the source and the target molecules are not part of the training set. Figure 4 presents the prediction accuracy against the training dataset size. To understand the results better, we segregate them into pathways of length 1, 2 and ≥ 3. The trends are similar across all lengths and the results saturate at around $\approx$35 000 reactions in the training dataset. As

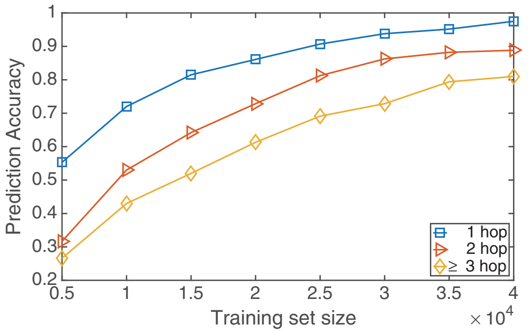


Fig. 4. Accuracy of pathway prediction against training dataset size, for pathways of varying lengths

expected, the accuracy is better for single length pathways since the search space is smaller. Theoretically, the search space increases exponentially by a factor of *d* with each hop, where *d* is the average degree of the RRN. For all three pathway lengths, the accuracy is higher than 80% at ≈35 000 training reactions and beyond. We also studied the scalability of our approach. Our results are detailed in Supplementary Results Section 2.2.

4 Discussion

Is it possible to synthesize molecule B from molecule A? Are there possible alternative routes to synthesize a molecule, other than the one followed by cells of living organisms? Why do organisms in nature choose a particular pathway to synthesize a metabolite, say pyruvate, from glucose? In this paper, we have developed a pathway prediction technique that can answer these questions. The proposed system is the first fully automated technique that can operate at the level of hundreds of thousands of reactions and answer queries in seconds. This level of sophistication is achieved through a graph mining based approach, which automatically mines cause-and-effect patterns of structural transformations from a training database of chemical reactions. These patterns are employed to construct an abstract representation of the reaction space in the form of a RRN. This abstract representation lies at the core of our ability to make rapid predictions, even on molecules that we have never seen before.

Many earlier studies have approached path finding in metabolic/chemical reaction networks; however, they typically fall short in one or more of the following: (i) they rely on the existence of query molecules in their database, or (ii) their pipeline involves the application of hand-curated rules or reactant–product mapping information, or (iii) they only work for specific classes of reactions. Using no more information than the molecular structure of every molecule in the reaction database, we have developed a powerful pipeline for predicting pathways between any two metabolites.

Our key findings fall into three categories. First, we have an efficient reactant–product mapping that is built on subgraph edit distance. It enables us to accurately track changes in chemical moieties across the entire spectrum of biochemical reactions. Next, we identified reaction signatures, which are essentially subgraphs necessary for the reactions to occur. Next, we embedded information about the *reaction centres* in a given metabolic network onto another network, the RRN. This novel representation enables us to predict a series of reactions (or, a pathway) connecting two metabolites, which may not even belong in the original reaction database.

We then proceeded to ask a more fundamental question about the organization of metabolic networks: What is the key underlying design principle of known metabolic pathways? For example, it is well-known that standard biochemical pathways do not represent shortest paths in the network—there are likely other constraints such as energetics in play. Other studies (Noor *et al.*, 2010) have shown that central carbon metabolism is a minimal walk between key precursor metabolites. We have here shown that across an assortment of pathways, nature appears to minimize the incremental biochemical change occurring, from the reactant to product, in every step of the reaction. By employing a heuristic built on this logic, we correctly recover a majority of pathways (see Supplementary Table S1) from carbohydrate, amino acid and fatty acid metabolism. This is a particularly novel aspect of our method, since it is conventionally believed that energy considerations predominate. Indeed, in certain cases, we observe that a different pathway is in use by nature, clearly owing to energy considerations. For example, the path from D-Mannose to L-Galactose in nature may be convoluted, owing to energy considerations: D-Mannose → GDP-mannose → GDP-L-galactose → β-L-Galactose → L-Galactose, even though a simple epimerization reaction may theoretically be possible. It is important to note that our graph formalism, coupled with our heuristic has enabled us make reliable predictions, even in the absence of important information such as manually curated reactant mapping or ΔG values for different reactions. It is also interesting to note that we accomplished this in central carbon metabolism, where the structural similarity of the molecules can potentially be more confounding, compared to specialized pathways.

We have also predicted retrosynthetic pathways to commercially important molecules such as 1,3-propanediol, naringenin, itaconate and artemisinic acid, and we compare favourably with previous methods such as MRE, FMM and RetroPath. Importantly, we are able to additionally predict pathways for compounds such as pentachlorophenol, which MRE and FMM are unable to. Our method also enables us to predict pathways for compounds not present in the training database, such as methyl nicotinate, methyl phenylacetate and methyl *p*-anisate.

Finally, we also demonstrated that our approach is very scalable. This is particularly important in the light of the fact that many studies have pointed out that our current understanding of microbial metabolism is rather myopic—many more organisms from diverse phyla need to be reconstructed, and even for many current metabolic network reconstructions, major gaps in the reactome are present (Monk *et al.*, 2014). A comparison with the BRENDA enzyme database also showed that only a third of the enzymatic activities in BRENDA are covered by currently available metabolic networks (Monk *et al.*, 2014). Given the significant imminent expansion in metabolic network databases, a scalable approach such as ours bears special significance. By synthetically expanding the KEGG database to about 150 000 reactions, we show that our approach is still very fast, able to answer queries in a matter of seconds.

Our method is not without limitations. In choosing to keep the input information as minimal as possible, to enable widespread applicability, we have chosen to leave out thermodynamics from the picture, often very essential for accurate predictions and ranking of pathways. Nevertheless, we demonstrate that even without thermodynamic information, we are able to recover a majority of natural biosynthetic pathways. Further, it is often difficult to obtain accurate measurements of changes in free energy, especially those which are organism-specific. Also, like most other similar approaches to predict reactions, the accuracy of our approach is limited by the accuracy of the reaction database, KEGG, in this case. KEGG also contains no information about the reversibility of reactions, and essentially assumes all reactions are reversible. However, it will be straightforward to integrate information from other databases such as MetaCyc; the scalability of our algorithm will be particularly handy in such scenarios.

In sum, we see three major contributions of our study. First, we define a robust reaction–product mapping method using subgraph edit distance, which is fast and reliable. This enables us to construct a novel representation of a database of chemical reactions in terms of a RRN that lends itself to rapid querying for pathways to synthesize even molecules that are not present in the original reaction databases. Next, we define a heuristic to perform searches on this network, by minimizing the extent of transformation in every reaction. Searching using this heuristic very effectively recovers known native pathways across organisms, and enables a realistic ranking of predicted alternate biosynthetic pathways. Finally, we demonstrate the ease with which we can provide solutions to retrosynthesis queries. Notably, our approach uses no information other than the chemical structure of the molecules in every individual reaction, and yet gives very accurate results and scales up to over a hundred thousand reactions.

Acknowledgements

The authors thank P. Kumar and A. Ravikrishnan for useful discussions and M. Narayanan for a critical reading of the manuscript.

Funding

This work was supported by the Indian Institute of Technology Madras [grant CSE/14-15/5643/NFSC/SAYN to S.R.].

Conflict of Interest: none declared.

References

Bhatter, P. *et al.* (2017) Elucidating the biosynthetic pathways of volatile organic compounds in *Mycobacterium tuberculosis* through a computational approach. *Mol. BioSyst.*, **13**, 750–755.

Cai, M. and Xun, L. (2002) Organization and regulation of pentachlorophenol-degrading genes in *Sphingobium chlorophenolicum* ATCC 39723. *J. Bacteriol.*, **184**, 4672–4680.

Carbonell, P. *et al.* (2012) Enumerating metabolic pathways for the production of heterologous target chemicals in chassis organisms. *BMC Syst. Biol.*, **6**, 10+.

Carbonell, P. *et al.* (2014a) Retropath: automated pipeline for embedded metabolic circuits. *ACS Synth. Biol.*, **3**, 565–577.

Carbonell, P. *et al.* (2014b) XTMS: pathway design in an eXTended metabolic space. *Nucleic Acids Res.*, **42**, W389–W394.

Carbonell, P. *et al.* (2011) A retrosynthetic biology approach to metabolic pathway design for therapeutic production. *BMC Syst. Biol.*, **5**, 122+.

Caspi, R. *et al.* (2012) The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic Acids Res.*, **40**, D742–D753.

Chen, J.H. and Baldi, P. (2009) No electron left behind: a rule-based expert system to predict chemical reactions and reaction mechanisms. *J. Chem. Inf. Model.*, **49**, 2034–2043.

Chou, C.H. *et al.* (2009) FMM: a web server for metabolic pathway reconstruction and comparative analysis. *Nucleic Acids Res.*, **37**, W129–W134.

Copley, S.D. (2009) Evolution of efficient pathways for degradation of anthropogenic chemicals. *Nat. Chem. Biol.*, **5**, 559–566.

Corey, E.J. and Wipke, W.T. (1969) Computer-assisted design of complex organic syntheses. *Science*, **166**, 178–192.

Ganter, M. *et al.* (2013) MetaNetX.org: a website and repository for accessing, analysing and manipulating metabolic networks. *Bioinformatics*, **29**, 815–816.

Hadadi, N. *et al.* (2016) ATLAS of biochemistry: a repository of all possible biochemical reactions for synthetic biology and metabolic engineering studies. *ACS Synth. Biol.*, **5**, 1155–1166.

Hadadi, N. and Hatzimanikatis, V. (2015) Design of computational retrobiosynthesis tools for the design of de novo synthetic pathways. *Curr. Opin. Chem. Biol.*, **28**, 99–104.

Hatzimanikatis, V. *et al.* (2005) Exploring the diversity of complex metabolic networks. *Bioinformatics*, **21**, 1603–1609.

He, H. and Singh, A.K. (2006) Closure-tree: an index structure for graph queries. In: *Proceedings of the 22nd International Conference on Data Engineering ICDE '06*, IEEE Computer Society, Washington, DC, USA, p. 38.

Kanehisa, M. *et al.* (2016) KEGG as a reference resource for gene and protein annotation. *Nucleic Acids Res.*, **44**, D457.

Kauzmann, W. (1959) Chemical specificity in biological systems. *Rev. Mod. Phys.*, **31**, 549–556.

Kayala, M.A. and Baldi, P. (2012) ReactionPredictor: prediction of complex chemical reactions at the mechanistic level using machine learning. *J. Chem. Inf. Model.*, **52**, 2526–2540.

Kotera, M. *et al.* (2014) Metabolome-scale prediction of intermediate compounds in multistep metabolic pathways with a recursive supervised approach. *Bioinformatics*, **30**, i165.

Kotera, M. *et al.* (2013) Supervised *de novo* reconstruction of metabolic pathways from metabolome-scale compound sets. *Bioinformatics*, **29**, i135–i144.

Kuwahara, H. *et al.* (2016) MRE: a web tool to suggest foreign enzymes for the biosynthesis pathway design with competing endogenous reactions in mind. *Nucleic Acids Res.*, **44**, W217–W225.

Latendresse, M. *et al.* (2014) Optimal metabolic route search based on atom mappings. *Bioinformatics*, **30**, 2043–2050.

Mavrovouniotis, M.L. *et al.* (1990) Computer-aided synthesis of biochemical pathways. *Biotechnol. Bioeng.*, **36**, 1119–1132.

McClymont, K. and Soyer, O.S. (2013) Metabolic tinker: an online tool for guiding the design of synthetic metabolic pathways. *Nucleic Acids Res.*, **41**, e113.

Medema, M.H. *et al.* (2012) Computational tools for the synthetic design of biochemical pathways. *Nat. Rev. Microbiol.*, **10**, 191–202.

Mithani, A. *et al.* (2009) Rahnuma: hypergraph-based tool for metabolic pathway prediction and network comparison. *Bioinformatics*, **25**, 1831–1832.

Monk, J. *et al.* (2014) Optimizing genome-scale network reconstructions. *Nat. Biotechnol.*, **32**, 447–452.

Moriya, Y. *et al.* (2010) PathPred: an enzyme-catalyzed metabolic pathway prediction server. *Nucleic Acids Res.*, **38**, W138–W143.

Noor, E. *et al.* (2010) Central carbon metabolism as a minimal biochemical walk between precursors for biomass and energy. *Mol. Cell*, **39**, 809–820.

Oh, M. *et al.* (2007) Systematic analysis of enzyme-catalyzed reaction patterns and prediction of microbial biodegradation pathways. *J. Chem. Inf. Model.*, **47**, 1702–1712.

Palmer, T. (2007) *Enzymes: Biochemistry, Biotechnology and Clinical Chemistry*. Horwood, Chichester.

Rahman, S.A. *et al.* (2005) Metabolic pathway analysis web service (pathway hunter tool at cubic). *Bioinformatics*, **21**, 1189–1193.

Rahman, S.A. *et al.* (2014) EC-BLAST: a tool to automatically search and compare enzyme reactions. *Nat. Methods*, **11**, 171–174.

Rahman, S.A. *et al.* (2009) Small molecule subgraph detector (SMSD) toolkit. *J. Cheminform.*, **1**, 12+.

Ro, D.K. *et al.* (2006) Production of the antimalarial drug precursor artemisinic acid in engineered yeast. *Nature*, **440**, 940–943.

Rosselló, F. and Valiente, G. (2004) Analysis of metabolic pathways by graph transformation. In: *Graph Transformations*. Springer, Berlin, Heidelberg.

Russell, S.J. and Norvig, P. (2003) *Artificial Intelligence: A Modern Approach*. 2nd edn., Pearson Education, India.

Sivakumar, T.V. *et al.* (2016) ReactPRED: a tool to predict and analyze biochemical reactions. *Bioinformatics*, **32**, 3522–3524.

Werpy, T. and Petersen, G. (2004) Top value added chemicals from biomass: volume I – results of screening for potential candidates from sugars and synthesis gas. Technical report.

Yamanishi, Y. *et al.* (2015) Metabolome-scale *de novo* pathway reconstruction using regioisomer-sensitive graph alignments. *Bioinformatics*, **31**, i161.

Zeng, Z. *et al.* (2009) Comparing stars: on approximating graph edit distance. *Proc. VLDB Endow*, **2**, 25–36.