SCIENTIFIC REPORTS

OPEN

Enumerating all possible biosynthetic pathways in metabolic networks

Aarthi Ravikrishnan[1,2,3], Meghana Nasre[4] & Karthik Raman[1,2,3]

Received: 21 February 2018
Accepted: 14 June 2018
Published online: 02 July 2018

Exhaustive identification of all possible alternate pathways that exist in metabolic networks can provide valuable insights into cellular metabolism. With the growing number of metabolic reconstructions, there is a need for an efficient method to enumerate pathways, which can also scale well to large metabolic networks, such as those corresponding to microbial communities. We developed MetQuest, an efficient graph-theoretic algorithm to enumerate all possible pathways of a particular size between a given set of source and target molecules. Our algorithm employs a *guided* breadth-first search to identify all feasible reactions based on the availability of the precursor molecules, followed by a novel dynamic-programming based enumeration, which assembles these reactions into pathways of a specified size producing the target from the source. We demonstrate several interesting applications of our algorithm, ranging from identifying amino acid biosynthesis pathways to identifying the most diverse pathways involved in degradation of complex molecules. We also illustrate the scalability of our algorithm, by studying large graphs such as those corresponding to microbial communities, and identify several metabolic interactions happening therein. MetQuest is available as a Python package, and the source codes can be found at https://github.com/RamanLab/metquest.

Genome-scale metabolic networks are very useful to understand the complex network of metabolic reactions happening inside cells[1–3]. Typically, a genome-scale metabolic network consists of thousands of reactions and metabolites, which capture vital metabolic pathways such as the biosynthesis of amino acids and lipids, ATP synthesis, as well as transport of molecules inside the cells. Over the recent years, the construction of such networks has increased tremendously, in part also due to the growing number of genomes sequenced[4]. There are several well-established constraint-based methods to analyse metabolic networks[5]; however, these methods require well-curated genome-scale metabolic models for making reliable predictions. The number of such well-curated models are very few[4,6,7], in comparison to the genome sequences available. Alternative approaches to analyse genome-scale models are based on network "topology"[8–10] or Boolean rules[11,12]. In the former, the metabolic networks are abstracted as networks or graphs, and graph-theoretic algorithms are employed to predict or infer metabolic pathways. The latter set of methods are based on definitions of Boolean functions for identifying if particular reactions can proceed or not. Both methods generate qualitative predictions and do not entail the requirement of well-curated genome-scale models.

Several graph-based methods, using naïve network expansion[8,13], atom–atom mapping, subgraph matching or stoichiometry[9,14–16] have been developed previously. These methods, regardless of the approach, aim to determine the route(s) of conversion between sets of source and target molecules. The algorithms based on network expansion traverse the graph depending on the availability of precursor compounds and determine different routes of conversions from the metabolic network. The input graph to these algorithms can be represented in different ways, such as a substrate graph[17] or a hypergraph[18], each of which captures information at different levels of complexity. Substrate graphs are one of the simplest forms of representation, where all metabolites participating in a reaction are connected to each other. Traversing these graphs using breadth-first or depth-first search to identify different routes of conversion often leads to erroneous paths such as a 2-step glycolysis, converting glucose to pyruvate (via ADP), since these could proceed using the connections between "side" metabolites. To avoid such spurious results, several tools, such as FMM (From Metabolite to Metabolite[19]); and Metabolic Tinker[20] use substrate graphs after excluding the connections from (high degree) "currency metabolites"[17] such as H^+, Water and

[1]Department of Biotechnology, Bhupat and Jyoti Mehta School of Biosciences, Indian Institute of Technology (IIT), IIT Madras, Chennai, 600036, Tamil Nadu, India. [2]Initiative for Biological Systems Engineering (IBSE), IIT Madras, Chennai, India. [3]Robert Bosch Centre for Data Science and Artificial Intelligence (RBC-DSAI), IIT Madras, Chennai, India. [4]Department of Computer Science and Engineering, IIT Madras, Chennai, India. Correspondence and requests for materials should be addressed to M.N. (email: meghana@iitm.ac.in) or K.R. (email: kraman@iitm.ac.in)

NADPH. However, the substrate graph representation fails to capture the fact that more than one reactant may be required for a reaction to occur.

To overcome these problems, algorithms were developed to operate on more informative bi-partite and hypergraph representations of metabolic networks[8,13,18,21,22]. These representations consider the participation of multiple compounds in a reaction to produce product(s), which is crucial while depicting metabolic networks. One such hypergraph-based algorithm, Rahnuma[18], performs a depth-first search to identify pathways that lead to the target metabolite from the source. In a few other studies[8,13], the metabolic networks are represented as bipartite graphs to first determine the *scope* of the starting seed metabolites. The synthesis pathways were then assembled by backtracking from the target to the source, which becomes computationally challenging due to the presence of branched and cyclic pathways in metabolic networks.

The problem with determining branched-pathways has been addressed in a few studies, which seek to use atom tracing. In one such method[23] BPAT-S (Branched Pathfinding using Atom Tracking and Seed pathways), *seed pathways* (linear) are first identified between the source and target compounds, based on the atom loss/gain. The branched pathways are then identified by finding the *linear pathways* between the metabolites contributing to this atom loss/gain. These pathways are then ranked based on the total number of reactions and the number of conserved atoms. ReTrace[24], another algorithm to find branched pathways, tries to combine linear shortest paths based on the transfer of a high fraction of atoms from source to target, thereby finding only the *k*-shortest paths.

Another class of methods use either information from atom-mapping, thermodynamics or structural transformation. For instance, the algorithm developed[9] defines a set of atom-mapping rules based on the possible chemical conversions and seeks to identify paths where the atom loss is minimum. The predictions of this algorithm are restricted to the first *k*-shortest paths. PathPred[14], another tool to predict metabolic pathways, performs subgraph matching on the graph generated from the KEGG RPAIR database. Similarly, RouteSearch[25], another algorithm based on branch-and-bound search, finds paths based on the mapped network generated from the atom mappings.

Another set of methods to study alternate pathways in metabolic networks are based on Elementary Flux Modes (EFMs), which are minimal sets of enzymes, operating in the correct direction, and are required for a given system to carry a steady-state flux[26,27]. EFMs have been widely used to study metabolic networks[28–31]; there are many tools that have been developed to study and evaluate EFMs, in different contexts[31–33]. However, the methods to enumerate EFMs, not only entail the requirement of well-curated genome-scale metabolic networks but also identify only the pathways at steady state. Such steady-state identification of pathways can limit the scope of analyses to *in vitro* experiments[34].

Despite the availability of several types of methods to understand and analyse metabolic networks, a simple and efficient method with minimum input for performing large-scale analyses is still lacking. Also, the current methods to identify pathways between the source and target molecules are restricted to smaller networks and do not enumerate long pathways, thereby restricting the scope of analyses. Although a few of these algorithms handle cyclic and branched pathways, they require additional information such as atom transfer, which may not be readily available from the semi-curated metabolic networks. Further, we find that many of these tools are web-based, and do not lend easily to large-scale analyses; a large fraction of these tools are also currently inaccessible (Supplementary Table S1).

Thus, there is a need for a method to find pathways, which (a) requires only the topology of the reaction network (rather than stoichiometry and atom mapping), (b) is simple and scalable to metabolic networks (especially those comprising more than one organism), (c) efficiently handles cyclic and branched pathways, and (d) examines multiple alternate routes of conversion. An important application of identifying and understanding such multiple alternate pathways is in metabolic engineering, where organisms are routinely engineered to produce a given molecule. To this end, we developed MetQuest, an efficient and scalable graph-theoretic algorithm, which exhaustively identifies all the pathways, or "sub-networks" between a given set of source and target metabolites using the input reaction network. In contrast to constraint-based approaches, our algorithm requires only the reaction network topologies which are readily available from draft reconstructions and does not necessitate the requirement of well-curated genome-scale models. Due to the nature of the implementation, we are able to successfully identify branched and cyclic pathways. In comparison to other graph-based algorithms, we observe better results, in terms of the completeness of the output pathways. Since we perform exhaustive enumeration, we not only recover the well-known pathways such as glycolysis and amino acid biosynthesis but also identify several diverse pathways such as those involved in biodegradation of pollutants such as catechol. Further, because of the scalability of our algorithm to operate on large metabolic networks, we are also able to demonstrate several metabolic exchanges happening in known natural and synthetic microbial communities.

Methods

In this section, we present a detailed overview of our algorithm MetQuest, broadly divided into two phases. The first phase involves a *guided* breadth-first search (BFS) of the metabolic network, to identify all the metabolites that can be reached from a given set of "seed" metabolites. In the second phase, we design a dynamic programming algorithm, which solves the non-trivial problem of assembling reactions into pathways to produce the metabolite of interest.

We begin by describing the input representation of the metabolic network. Any given metabolic network can be represented as a directed bipartite graph $G(M, R, E)$, where M is the set of metabolites in the metabolic network, R is the set of reactions and E is the set of edges. Directed edges connect metabolites ($m_i \in M$) to a reaction node ($r_j \in R$) or a reaction node to product metabolites. The reversible reactions in the metabolic networks are denoted by two separate reaction identifiers, representing the forward and the reverse reactions, respectively. We construct the directed bipartite graph G of microbial communities (consisting of more than one metabolic network) by connecting the graphs of individual organisms through a common extracellular medium, based on the overlapping set of exchange reactions, as described elsewhere[35]. The non-common exchange reactions are connected only to the extracellular environment. Such a bipartite representation disallows invalid conversions as may be interpreted from substrate graphs and helps in generating valid paths with biologically *meaningful* conversions[36]. The input to MetQuest is a directed bipartite graph G derived from a given metabolic network, a set S of seed metabolites, a set T of target metabolites and an integer β which bounds the *size* of any pathway generated by MetQuest. Note that the set S of seed metabolites includes the source metabolite(s) as well as molecules such as co-factors and co-enzymes that are commonly present in any cell. Supplementary Tables S2–S5 enumerate the seed metabolites we have used in our analyses. Below we define formally a pathway and the size of a pathway, which are crucial to our algorithm:

Definition 1 (Reachable metabolite m**):** A metabolite m is reachable from a set S if either m is in the set S or there is a reaction r in the reaction network whose output is m and every input of r is producible.

Definition 2 (Branched pathway producing m**):** An S-to-m pathway R' is a set of reactions such that m is the output of at least one reaction in R' and every input of every reaction in R' is producible from S. Throughout the paper, we will use the term pathway or sub-network interchangeably.

Definition 3 (Cyclic pathway producing m**):** A cyclic pathway R', from S to m is a set of reactions where m, which is the output of at least one reaction in R' is used in its own production by another reaction in R'.

Definition 4 (Size of a pathway): The size of a pathway R' is the cardinality of the set R', i.e. the number of reactions in the set R'.

The goal of MetQuest is to identify all pathways of size at most β that produce the target metabolites from the seed metabolites. We now describe the two phases of our algorithm.

Phase 1: Guided BFS. BFS is a classic graph traversal technique that visits all the nodes of a given graph, starting at a source node, in a breadth-first fashion. BFS employs a queue of vertices, where newly discovered vertices are enqueued, to be processed at a later stage. A complete description of the BFS algorithm can be found elsewhere[37,38]. We modify the standard BFS by *guiding* it, based on the availability of precursor metabolites. Starting with the set of seed metabolites S, the algorithm first finds all the reactions from the set R, whose precursor metabolites are in S. Such reactions are marked "visited" and added to the visited reaction set R_v. The metabolites produced by these reactions, m_c, are then added to S. The traversal continues in a *breadth-first* manner, incrementally adding triggerable reactions to the BFS queue. The expansion stops when there are no further reactions that can be visited. During the expansion, a reaction node is labelled as *stuck*, if it does not (yet) have the necessary precursors in S. Such reactions are automatically triggered if the precursor metabolites are produced at any later stage. A formal description of this phase of the algorithm can be found in Supplementary Algorithm S1. The traversed graph consists of all reactions that can be *visited*.

At the end of the traversal, we obtain the scope $M_s \supseteq S$ and the set of visited reaction nodes, R_v. The scope M_s comprises all metabolites that can be produced from the seed set S, in the given metabolic network. We also obtain the minimum number of steps to reach any metabolite m and reaction r, starting at the seed metabolite set S, denoted as ℓ_m and ℓ_r, respectively. We note that the value ℓ_m for a metabolite m (analogously ℓ_r for a reaction r) is the shortest path distance in terms of the number of edges from the seed set S to the metabolite m (reaction r respectively). Thus, the number ℓ_m does not necessarily indicate the exact number of reactions required to produce the metabolite m. Instead, we only leverage it for algorithm optimisation (see Supplementary Methods § 4.1).

This process of graph traversal resembles the ideas of network expansion[8], and forward propagation[39] reported earlier. The former method seeks to identify the synthesising capacity of the metabolic network using the input seed set of metabolites. The latter aims to identify the minimal precursor sets of seed metabolites that are required to produce a given set of target metabolites. This step in our algorithm also derives the information about the scope of metabolites from the given metabolic network; it, however, distinguishes itself by making a systematic note of the visited and stuck reaction nodes, which are exploited at a later stage for efficient and exhaustive enumeration of biosynthetic pathways.

Phase 2: Generation of pathways. MetQuest uses a recursive dynamic programming formulation for computation of sub-networks. However, it avoids repeated recursive calls by *memoising* the precomputed values. The algorithm steps are listed below and the pseudo-code can be found in Algorithm 1 and Algorithm 2. Our algorithm maintains a table of size $|M_s| \times \beta$ and initialises all the entries to $\perp$, indicating that we do not (yet) know about the pathways for every metabolite. Recall that M_s and β denote the *scope* of seed metabolites and the maximum number of reactions we allow in a pathway (the "size cut-off") respectively. We start filling the table entries by first considering the seed metabolite set S. For every seed metabolite $m \in S$, the entry in corresponding cell $Table(m, 0) = \varnothing$, indicating that no reaction is required to produce it; for every metabolite $m \in M_s \backslash S$, the entry $Table[m][0]$ remains as $\perp$ (Lines 1–5). The goal of MetQuest is to populate the entries of the *Table*. At the end of the algorithm, for any metabolite $m \in M_s$ and an integer k (where $0 \leq k \leq \beta$), the entry $Table[m][k]$ is a set of pathways or $\perp$. If the entry is not $\perp$, each pathway in the set $Table[m][k]$ is of size k and produces the metabolite m starting from the seed metabolite set. Further, $Table[m][k] = \perp$ implies that m cannot be produced starting from the seed metabolite set S using exactly k reactions.

Algorithm 1. MetQuest

```
1:  for each metabolite $m \in M_s$ do
2:      if $m \in S$ then
3:          $Table[m][0] = \emptyset$
4:      else
5:          $Table[m][0] = \perp$
6:  for each reaction $r \in R_v$ whose inputs are seed metabolites do
7:      for each output $m$ of $r$ do
8:          $Table[m][1] = Table[m][1] \cup \{r\}$
9:  for $k = 2 \ldots \beta$ do
10:     for each reaction $r \in R_v$ do
11:         Let $n$ denote the number of inputs to $r$
12:         Let $m_1, m_2 \ldots m_n$ denote the inputs to $r$
13:         for $\ell = k - 1 \ldots n \times (k - 1)$ do
14:             if $\ell \le n \times (k - 2)$ then
15:                 $t = \lfloor \frac{\ell}{k-1} \rfloor$
16:                 for $\ell' = 1 \ldots t$ do
17:                     Fix $\ell'$ inputs that take a value of $k - 1$
18:                     This can be done in $\binom{n}{\ell'}$ ways
19:                     $\mathcal{P} = \textbf{generatePartitions}(\ell - (k-1)\ell', n - \ell', k - 1)$
20:             else
21:                 $\mathcal{P} = \textbf{generatePartitions}(\ell, n, k - 1)$
22:             for each $p \in \mathcal{P}$ do
23:                 $Table = \textbf{populateTable}(r, p)$
```

Note: k denotes the maximum size of a pathway that we intend to find. At the end of the algorithm $Table[m][k]$ contains all possible pathways of size k that produce m from the seed metabolite set S. If at the end of the algorithm, an entry $Table[m][k] = \perp$, then it implies that m cannot be generated using exactly k reactions starting at the seed metabolite set S. See Supplementary Methods §4.2 for an explanation about generatePartitions function.

Algorithm Steps.

1. In the first iteration, we find all the reactions $r \in R_v$, whose inputs are only seed metabolites. For every output metabolite m produced by r, we fill $Table(m, 1)$ with the reaction that produced m (Algorithm 1, Lines 6–8).

Algorithm 2. populateTable(p)

```
function POPULATETABLE$(r, p)$
// $Table$ is global. Recall that $p$ is an n-tuple of integers obtained from Algorithm 1
    if $Table[m_q][p_q] = \perp$ for any $q \in 1 \ldots n$ then
        return
    $\mathcal{SRS} = Table[m_1][p_1] \times Table[m_2][p_2] \times \ldots \times Table[m_n][p_n]$;
    for every $\mathcal{RS} \in \mathcal{SRS}$ do
        $\mathcal{RS} = \mathcal{RS} \cup \{r\}$
        $j = |\mathcal{RS}|$
        for each output $y$ of $r$ do
            $Table[y][j] = Table[y][j] \cup \mathcal{RS}$
end function
```

2. Our algorithm fills *Table* column-wise and the `for` loop (Algorithm 1, Line 9) iterates over the columns ranging from 2 to β. For any fixed column value k, the `for` loop (Algorithm 1, Line 10) considers every reaction r in the set R_v. We now propose that a pathway of size k, consisting of a reaction $r \in R_v$, can be constructed by merging the pathways that generated the input metabolites of r.
3. Let $m_1, m_2, \ldots, m_n$ denote the n inputs to reaction r. Since our goal is to construct a pathway of size k containing the reaction r, we note that the sum of the sizes of the pathways that generate inputs to r must be at most $k - 1$ (adding one for the reaction r to this merged pathway will make the size k). Thus, a straightforward strategy would be to consider all possible ways of achieving the sum $k - 1$ using exactly n non-negative integers. The permissible values for each of these n integers range from 0 to $k - 1$. For example if $n = 2$ and $k = 3$, then all possible ways to generate a sum of 2 are $(0, 2)$, $(1, 1)$, $(2, 0)$. We call any single such possible way (for example $(1, 1)$) as a partition of the integer k.

4. Now, assume that we have fixed a set of n non-negative integers $(p_1, p_2, \ldots, p_n)$, which sum up to $k - 1$. Note that for any i, the value p_i denotes the size of the pathway for the i^{th} input m_i that will be used in this combination. A possible way to generate *one* pathway of size k is to pick a pathway from $Table[m_i][p_i]$ for $1 \leq i \leq n$, merge these pathways (that is take a union of the reactions in these pathways) and finally add the reaction r to this set. However, we show a simple example illustrating why such an approach is not guaranteed to generate a pathway of size k. Consider a two-input reaction r, where the inputs are m_1 and m_2. Say, the value of $k = 3$. The above listed approach will consider all possible partitions of the integer 2, which are $(0, 2)$, $(1, 1)$ and $(2, 0)$. Let us assume that our metabolic network has a reaction r', which produces both m_1 and m_2 using only the seed metabolites. In such a case, $Table[m_1][1]$ and $Table[m_2][1]$ both contain the pathway $\{r'\}$. Now note that merging these "two" pathways and adding the reaction r creates a pathway $\{r', r\}$ has size 2 instead of the intended size 3 (see Supplementary Methods § 4.3).
5. To address the above issue, our algorithm not only generates the sum $k - 1$ using all possible partitions, but also generates all sum values all the way up to $n \times (k - 1)$. This is achieved in the `for` loop of Line 13 of our algorithm. Note that although the sum-value is larger than $k - 1$, the integers used in any partition can take values only up to $k - 1$. This is because when our algorithm is at a particular column k, all entries in the *Table* up to the columns $k - 1$ are fully generated. However, for a fixed value of k, our algorithm may update *Table* entries in the column values larger than k.
6. The `for` loop from Line 13–21 (Algorithm 1) generates a set of partitions $\mathcal{P}$. As mentioned earlier, a partition $p \in \mathcal{P}$ is an n-tuple and the i^{th} entry p_i in the tuple denotes the size of the pathway to be used for the i^{th} input of the reaction r. We defer the details of the generation of the set $\mathcal{P}$ to Supplementary Methods § 4.2 and 4.3.
7. For every $p \in \mathcal{P}$, we invoke the function **populateTable** (Algorithm 1, Lines 22–23) for the reaction r and the partition p. Let y be any output of the reaction r. The goal of **populateTable** is to populate the entries of the form $Table[y][j]$ for values of j ranging from k to $(n \times (k - 1)) + 1$.
8. For a particular partition $p = (p_1, p_2, \ldots, p_n)$ and a reaction r, we fetch the entries $Table[m_i][p_i]$ for $i = 1 \ldots n$. Note that $Table[m_i][p_i]$ itself is a set of pathways each of size p_i. We now compute the set $\mathcal{SRS}$ by taking a cross product of these table entries (Algorithm 2, Line 5).
9. Our algorithm iterates over every pathway or reaction sub-network $\mathcal{RS} \in \mathcal{SRS}$ (Algorithm 2, Line 6). Next, we add the reaction r to the set $\mathcal{RS}$ (Algorithm 2, Line 7). We denote by j, the size of the new pathway thus produced. Finally, we add the newly generated pathway to the appropriate table entry $Table[y][j]$ and complete the call to Algorithm **populateTable**.

We note that the output pathways for every metabolite are generated by considering the sub-networks of *all* the input metabolites of a reaction. Due to this, any pathway generated by our algorithm is *complete*, i.e. the reactants required by the reactions constituting the pathway are all producible. Further, since for any metabolite, we perform a union of sets of reactions, we avoid repeated generation of the same reaction sets. Due to this, we automatically report only the first occurrence of cyclic pathways. The demonstration of our algorithm on a toy-network and a cyclic pathway can be found in Supplementary Methods § 4.4 and § 4.5 respectively. Further implementation details and the formal proof of correctness for our algorithm can be found in Supplementary Methods § 4.6 and § 4.7, respectively. MetQuest is available as a Python package at PyPI, and the source codes are available at https://github.com/RamanLab/MetQuest. Supplementary Methods § 4.6 also illustrates various analyses that can be performed with MetQuest such as identifying the most commonly occurring exchange metabolites in a microbial community and finding the most different pathways by calculating the Jaccard index[40] using the constituent reactions.

Results

In this section, we show that our algorithm can recover sub-networks of different types, including the well-known glycolysis and amino acid synthesis pathways. We also compare our output sub-networks with those generated by other algorithms and demonstrate that MetQuest produces better results since it generates *complete* pathways. Further, we showcase the ability of MetQuest to identify diverse pathways involved in biodegradation of an important industrial pollutant. Finally, we show that MetQuest scales well to large networks, and is, therefore, a powerful tool to predict and understand metabolic exchanges in microbial communities.

MetQuest exhaustively identifies multiple pathways in metabolic networks.

Genome-scale metabolic networks catalogue numerous metabolic pathways happening inside a cell. Identifying these pathways helps us to better understand the level of redundancy in the cells for producing key metabolites. To this end, we applied MetQuest on well-curated networks to identify and understand pathways between different reactants and products.

Central carbon metabolism. We identify well-known biochemical pathways such as glycolysis using the genome-scale metabolic model of *E. coli* iJO1366[41]. We constructed the corresponding directed bipartite graph, consisting of 12,974 edges and 5,659 nodes, and specified pyruvate as the target node. We included glucose from extracellular environment ("glc-D_e"), co-factors, co-enzymes and the energy currencies in the seed metabolite set S (Supplementary Table S2) and found all the sub-networks within a size cut-off β of 15.

We successfully recovered the well-known glycolysis pathway and, in addition, due to the exhaustive enumeration, we could also identify 4,787 paths from the seed set of metabolites S to the target of varying sizes. Of all these paths, 1007 of them (of $\beta \leq 15$) use glucose ("glc-D_e") as the starting metabolite, while the rest of the paths comprise compounds formed by seed metabolites reacting amongst themselves.

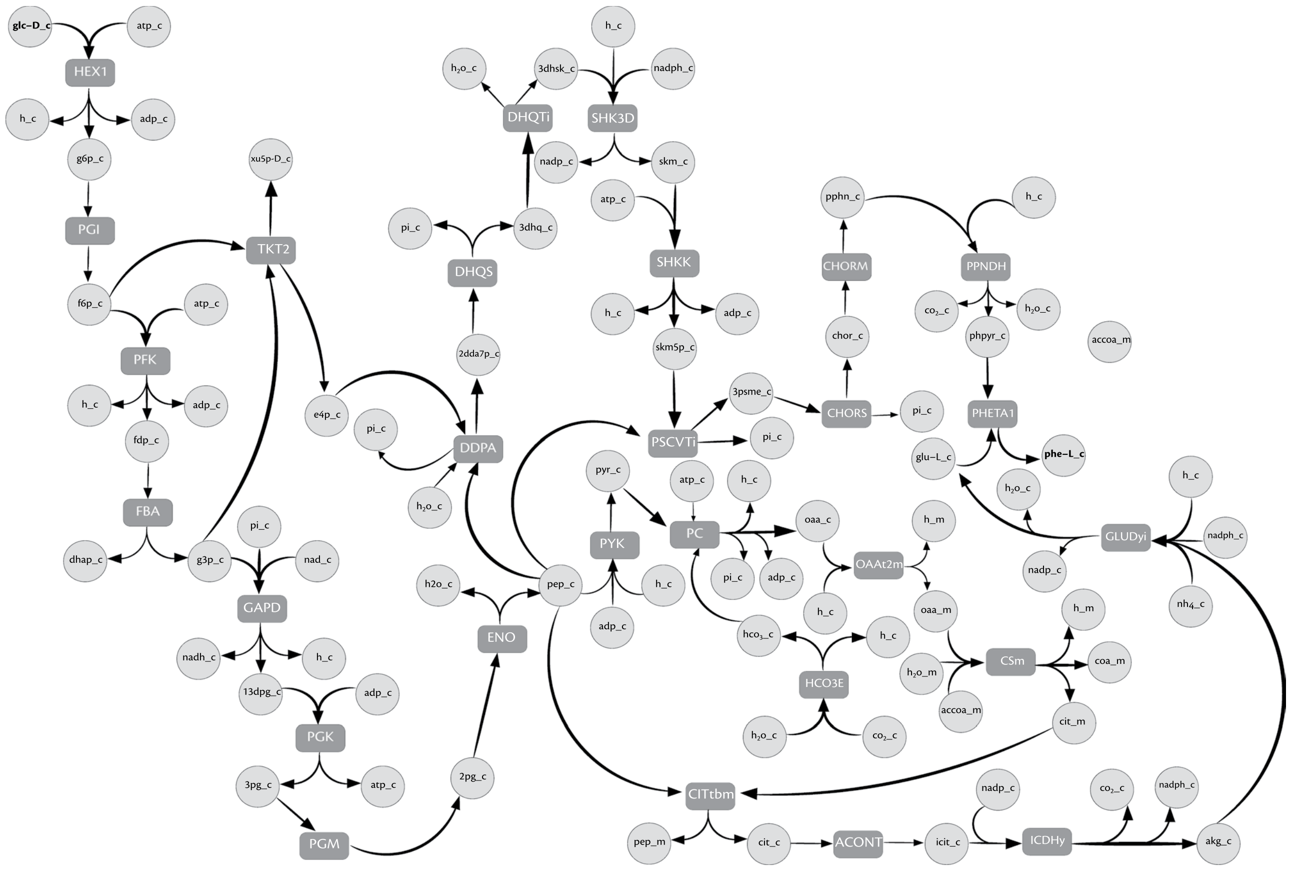

Figure 1. Sub-network with a size cut-off β of 28 between D-glucose and L-phenylalanine in *S. cerevisiae iMM904*. Dark gray rectangles represent reactions, light gray circles represent metabolites. The nomenclature of reaction and metabolite names are consistent with *i*MM904 genome-scale metabolic model. The source and the target metabolite is glc-D_c and phe-L_c respectively (shown in boldface and italics). We note that the synthesis of L-phenylalanine involves several metabolites from different compartments. The reactions in this pathway and the seed metabolites can be found in Supplementary Results § 5.1 and Supplementary Table S3 respectively.

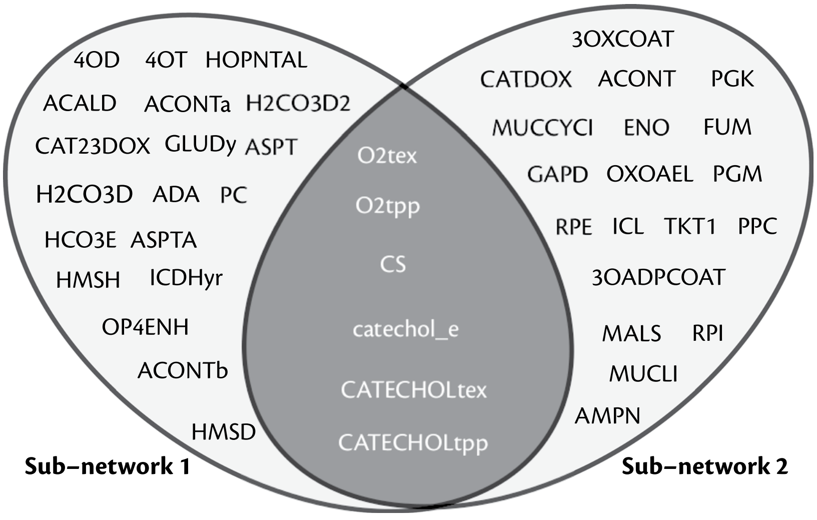

Figure 2. Overlap of the two most different pathways that degrade catechols in *Pseudomonas putida*. The two ovals represent reactions present in the two most different sub-networks identified. The intersection of the reactions found in the two sub-networks (shaded dark grey) pertains to the uptake of catechol. All the reaction identifiers follow the nomenclature from[46]. These reactions can be found in Supplementary Results § 5.2.

Source	Target	Size	Output sub-network	Comments
L-Arginine (C00062)	L-Citrulline (C00327)	2	R00551, R00665	Matches with ATLAS and FMM
Pyruvate (C00022)	Itaconate (C00490)	4	R02491, R00209, R00237, R02405	Matches with FMM, FMM does not report R00209 which produces C00024 – required by R02405†
Pyruvate (C00022)	Itaconate (C00490)	5	R00351, R02243, R00209, R00217, R01325	Matches with FMM, FMM does not report R00351 which produces C00036 – required by R00351†
L-Tyrosine (C00082)	Naringenin (C00509)	5	R02446, R00737, R01616, R01613, R06641	Matches with FMM, FMM does not report R06641†
L-Phenylalanine (C00079)	Resveratrol (C03582)	5	R01616, R00697, R02253, R06641, R01614	Matches with FMM, FMM does not report R06641 which produces malonyl-CoA required by R01614†
Mevalonic acid (C00418)	Amorpha-4,11-diene (C16028)	7	R01658, R03245, R02245, R01121, R01123, R07630, R02003	No paths found by FMM, ATLAS, however it is natively found in *S. cerevisiae*[51].
D-Erythrose 4-phosphate (C00279)	3-Amino-5-hydroxy-benzoate (C12107)	7	—	No paths reported by ATLAS, FMM and our algorithm

Table 1. The sub-networks produced between different source and target molecules for a given size. These results were obtained on the graph constructed using reactions in the KEGG database. We denote reactions and compounds using their respective KEGG IDs. It is interesting to note that our algorithm is able to recover the well-known pathway reported in *S. cerevisiae*, while the other algorithms fail to do so. Also, there exists no pathways of size 7 between C00279 and C12107, which is being consistently reported by all the methods. More details about the sub-networks can be found in Supplementary Results § 5.3. Note that the source metabolites are included in the seed metabolite set. †Note that we did not find any pathways in ATLAS, when queried for pathways using no *novel reactions*. These novel reactions, integrate KEGG metabolites into *novel enzymatic steps* and are present only in the ATLAS database.

Pyruvate								
	E. coli **core model**[58]		***L. lactis***[59]		***P. stipitis*** **iBB804**[60]		***S. cerevisiae*** **iMM904**[45]	
Pathway size	**Branched**	**Cyclic**	**Branched**	**Cyclic**	**Branched**	**Cyclic**	**Branched**	**Cyclic**
≤10	4	5	1	—	—	—	1	—
≤15	4	7	3	4	17	3	54	5
≤20	4	7	4	9	39	8	418	10
≤25	4	7	4	9	56	13	1783	15
Oxaloacetate								
	E. coli **core model**[58]		***L. lactis***[59]		***P. stipitis*** **iBB804**[60]		***S. cerevisiae*** **iMM904**[45]	
Pathway size	**Branched**	**Cyclic**	**Branched**	**Cyclic**	**Branched**	**Cyclic**	**Branched**	**Cyclic**
≤10	1	3	—	—	—	—	—	—
≤15	1	5	3	—	9	1	26	3
≤20	1	5	4	—	43	6	238	8
≤25	1	5	4	—	165	11	1475	13

Table 2. Number of pathways found for each target within the given cut-off β from the seed metabolite set S (which also includes the seed metabolite D - Glucose). We observe many pathways of size 20 for pyruvate in all the models but there are no pathways of size 10 that produce oxaloacetate in most of the models.

These observations raise an important question: “Why should so many pathways potentially exist inside the cell in addition to glycolysis, which is the most preferred pathway?” To answer this question and understand the differences between each of these pathways, we studied the similarity between the obtained sub-networks, using the Jaccard index. Of $\binom{1007}{2} \approx 5 \times 10^5$ combinations of sub-networks analysed, we find only 202 (0.03%) sub-networks are similar with Jaccard values between 0.93 and 0.98, while 1,08,388 (21.4%) sub-networks are much different with Jaccard values between 0.03 and 0.13 (Supplementary Figure S1).

The former set of sub-network pairs, with higher Jaccard values, can be understood in light of the presence of multiple transporters, which are either used to transport compounds from the environment or between different compartments of the cells. These transporters function under varied environmental conditions, where they employ different mechanisms to transport glucose inside the cell[42]. The latter set, with lower Jaccard values, point towards the existence of alternate pathways that produce identical precursor metabolites to synthesise pyruvate. These predominantly include the methylglyoxal pathway, the Entner-Doudoroff pathway and the pentose-phosphate pathway. The existence of many alternate pathways points towards the extent of redundancy in metabolic networks, which helps the organism salvage gene losses[43]. Further, some of these alternate pathways were also shown to outperform the canonical pathways under different sets of physiological conditions[44]. Identification of such alternate pathways by MetQuest through an exhaustive enumeration not only showcases its ability to correctly identify pathways on well-studied networks, but also renders MetQuest a viable tool to identify pathways from metabolic networks about which very little is known.

We also determined sub-networks between different metabolites in a compartmentalised model of *S. cerevisiae i*MM904[45]. As there are additional transport reactions between the compartments, we used a higher cut-off $\beta = 30$ and found the sub-networks between cytosolic D-Glucose ("glc-D_c") and amino acids such as L-phenylalanine. We assumed a seed metabolite set with 29 metabolites (Supplementary Table S3), consisting of the co-factors and co-enzymes in different compartments, and searched for sub-networks to 638 metabolites ($|M_s|$ was 681). Specifically, we identified 56 sub-networks from "glc-D_c" to target, whose size is less than 30. From the sub-networks, we find that many of these amino acids are derived from the intermediates of the central carbon metabolism. Also, the sub-networks producing aromatic amino acids involve metabolites produced in different compartments of the cell, thereby requiring longer steps for conversion(Fig. 1). Sub-networks of size $\beta = 28$ can be found in Supplementary Results § 5.1.

Identifying diverse paths for catechol biodegradation. Phenols and catechols are among the primary pollutants in the effluent from several industries, such as the chemical, textile and steel. So far, many studies have used *Pseudomonas putida* for bioremediation of such pollutants[47]. We sought to identify the pathways that render *P. putida* unique and tailored for this application. To this end, we investigated pathways involved in the metabolism of such compounds, which are otherwise harmful to many other micro-organisms. Further, we explored if the breakdown of catechols could directly lead to energy producing pathways. Hence, we identified all the pathways that start from catechols and lead to the intermediates of the tricarboxylic acid (TCA) cycle. Since catechols require complex mechanisms to be degraded to smaller intermediates, we used a higher cut-off β of 25. Using this, we found 1387 different paths of sizes up to 25 that degrade catechols to fumarate. Interestingly, our algorithm could identify the two most different mechanisms of ortho- and meta-cleavage used by *P. putida* to degrade catechols to the intermediates of the TCA cycle (Fig. 2), as reported[48]. These sub-networks can be found in Supplementary Results § 5.2.

MetQuest excels in comparison with other algorithms. To benchmark our algorithm, we compared our results with those obtained from some of the already existing path-finding methods such as FMM[19], ReTrace[24] and the pathways generated in ATLAS database[49]. Specifically, we determined paths/pathways between different source and target molecules, for multiple size cut-offs β. We constructed the bipartite graph corresponding to all reactions in the KEGG database[50]. For all the test-cases, we used the previously listed restricted compounds[20] as the seed set of metabolites.

We find that MetQuest performs better, in terms of the number and the completeness of the pathways generated, i.e. the output sub-networks are *complete*, in that they have all the reactions necessary to produce every reactant in that pathway. From the sub-networks (Table 1), we observe that the smaller pathways of size 2 completely match with those generated by the other algorithms. However, in many cases, we identify longer pathways, since these involve metabolites generated by branched pathways. It is interesting to note that our algorithm was able to correctly identify the already reported pathway between C00418 (Mevalonic acid) and C16028 (Amorpha-4,11-diene)[51], which was not identified by the other algorithms. All the sub-networks reported in Table 1 can be found in Supplementary Results § 5.3.

MetQuest scales well to large genome-scale and community metabolic networks. We examined the performance of MetQuest by finding sub-networks of varying size cut-offs ($\beta = 10, 15, 20, 25$) between glucose and other key metabolites involved in the central carbon metabolism on four genome-scale metabolic networks of different sizes (Table 2). We used a uniform set of seed metabolites consisting of essential co-enzymes and co-factors for all the simulations (Supplementary Table S4) and determined the sub-networks of sizes 10, 15, 20 and 25. During each simulation, we measured the time taken and the number of sub-networks generated for the target metabolites pertaining to the given size cut-off β. We carried out all the simulations on an Intel Core i7-2600 Desktop with 24GB RAM, running Ubuntu 16.04 LTS. Although the running time of the algorithm is exponential with the size of the model (Supplementary Figure S3), MetQuest does not fail to generate pathways, even at higher values of β.

Beyond individual networks, it is also interesting to identify sub-networks in communities of organisms and understand their metabolic interactions, which are known to play a critical role in determining the stability of microbial communities[52]. However, *community metabolic networks* are much larger in size, presenting many more challenges for path-finding. Analysing such large graphs of microbial communities demands a scalable and efficient algorithm, which requires only minimum information such as network topology. Further, the algorithm should be capable of identifying longer pathways to capture many metabolic exchanges happening within a community. The existing algorithms, such as OptCom[53] and cFBA[54], are based on constraint-based techniques and require well-curated metabolic models. Also, they do not directly lend themselves to identify metabolic interactions in draft network reconstructions. To this end, we use MetQuest on *joint graphs* (metabolic networks) of multiple organisms and identify sub-networks between different metabolites. Specifically, we construct the *community bipartite graph* of a microbial community by considering a common extracellular space through which metabolite exchanges can happen. Further, we analyse the sub-networks generated by MetQuest and show that it can correctly recover previously reported metabolic interactions.

MetQuest correctly predicts metabolic exchanges in a synthetic E. coli community. To study the metabolic interactions in *Escherichia coli*, computational analyses were performed on genetically modified *E. coli* strains[55]. Specifically, two different *E. coli* strains with knockouts of b2276 and b3708 genes were modelled together, and the metabolic interactions were predicted. We simulated these gene knockouts by removing the reactions catalysed by these genes from the genome-scale metabolic model[2], after considering the Gene-Protein-Reaction relationships. We then constructed the community bipartite graph and applied MetQuest to predict the metabolic exchanges between these two strains. We assumed seed metabolites, including glycolate (as given[55]) and applied MetQuest to find all sets of sub-networks to every metabolite within the scope of glycolate, pyruvate and both these sources independently, with a size cut-off $\beta = 20$. For this metabolic network with the number of nodes (metabolite $|M|$ and reactions $|R|$) =10113 and the number of edges $|E| = 23492$, MetQuest took 119.62 seconds to generate all possible sub-networks of size $\beta = 20$ from the given set of seed metabolites ($|S| = 49$).

From the sub-networks generated, we determined the metabolites that can be potentially exchanged between the strains. We were able to recover the reported acetate and formate exchanges happening between the two strains. In addition, we observe that acetate (from $\Delta b2276$) participates in the production of several important target metabolites in $\Delta b3708$ such as L-glutamine, inosine and several other co-enzymes of central carbon metabolism. We also found other metabolites such as alpha-ketoglutarate, ethanol, acetaldehyde exchanged between the organisms, which can be potential candidates for experimental verification (Supplementary Results § 5.4).

Predicting novel interactions in community metabolic networks. To illustrate the performance of MetQuest on a larger network, we computed multiple sub-networks in an experimentally demonstrated three-member microbial consortium[56], and determined the metabolic interactions. Towards this, we used the genome-scale metabolic models of *Clostridium cellulolyticum* (cc), *Desulfovibrio vulgaris* (dv), and *Geobacter sulfurreducens* (gs) from the Path2Models database[57], which hosts metabolic networks generated through automatic pipelines. We constructed the three-member community bipartite graph by connecting the genome-scale metabolic models through their common exchange reactions. We defined the seed metabolite set S, consisting of essential salts, co-factors, co-enzymes and a set of *tRNA* molecules. We also added cellobiose to this set (as reported[56]) and chose to determine sub-networks with a size cut-off $\beta = 20$, to all metabolites within the scope M_s of seed metabolites. For this metabolic network with number of nodes (metabolite $|M|$ and reactions $|R|$) =14521, $|E| = 29939$, MetQuest took ≈10 minutes to generate all possible sub-networks of size $\beta = 20$ from the given set of seed metabolites ($|S| = 373$). To determine the metabolic exchanges, we analysed every sub-network of 1,599 metabolites (M_s) with a special focus on sub-networks involving exchange metabolites. From this analysis, we found many sub-networks having at least one exchange reaction between the organisms. We observed that the proposed acetate and ethanol exchanges[56] predominantly led to the production of amino acids such as L-serine, L-leucine, L-aspartate and L-valine (in gs) and L-threonine and L-glycine (in dv) respectively, which contribute to the biomass, thereby enabling a stable microbial consortium.

Discussion

With the increasing number of draft genome-scale metabolic reconstructions, there is a pressing need for efficient algorithms to analyse these metabolic networks and generate useful predictions. It is particularly important that these algorithms perform efficiently with minimum information such as the reaction topology, and also identify pathways in large (multi-)genome-scale metabolic networks.

To this end, we propose MetQuest, a scalable and efficient graph-based algorithm for identifying all possible metabolic pathways in genome-scale metabolic models. Specifically, we focus on exhaustively determining all alternate pathways (of a particular size) between a set of seed and target metabolites. We also demonstrate the application of our algorithm by identifying multiple pathways involved in several important parts of metabolisms such as the central carbon metabolism and the amino acid metabolism. Further, we show its usefulness in determining several diverse pathways, which take part in catechol degradation. Besides its applications on the metabolic networks of individual organisms, MetQuest, due to its scalability, can be applied to study larger and more complex networks.

A number of graph-based methods to find pathways in metabolic networks have been developed so far. These methods convert the metabolic networks either into substrate graphs, bipartite graphs or hypergraphs. However, these methods are well-suited only for smaller networks and lower cut-offs. For instance, MetaPath[13] tries to backtrace the pathway from the target to the source based on the information obtained from the scope calculation. Since the metabolic networks contain many branched pathways, identifying the routes of conversion by back tracing becomes computationally expensive, and can break down rapidly at higher cut-offs. Another method, Rahnuma[18], tries to solve the problem of path prediction on genome-scale metabolic networks by abstracting these networks into hypergraphs. Rahnuma, based on depth-first search on hypergraphs, seeks to obtain pathways between two metabolites. However, the performance of Rahnuma on metabolic networks was demonstrated only for shorter path-lengths of 6, and it may not be applicable for identifying longer pathways, such as those involved in the biosynthesis of aromatic amino acids from simple sugars. Moreover, the pathways are computed based on a condition that the metabolite can be used only once in any pathway, which inherently fails to capture cyclic pathways. MetQuest, on the other hand, can identify branched and cyclic pathways, and can perform well on larger metabolic networks for much longer cut-offs, thereby extending its application to analysing microbial communities. Further, MetQuest does not entail the necessity of well-curated metabolic networks, as demanded by enumeration methods based on elementary modes. This renders MetQuest a useful tool to identify pathways and perform large-scale analyses using draft reconstructions. The identified pathways can be further analysed through elementary mode based techniques.

We have also demonstrated the major strengths of MetQuest: in multiple examples we considered, of individual metabolic networks and real-world microbial communities, MetQuest was able to identify multiple pathways and correctly predict the metabolic exchanges/interactions taking place. MetQuest was able to correctly predict the acetate exchange happening between the two genetically modified *E. coli* strains. Further, MetQuest also predicted novel interactions involving amino acids such as L-cysteine in the previously reported microbial community[56].

However, as with any modelling technique, MetQuest also has its limitations. Firstly, the predictions from MetQuest rely heavily on the quality of the underlying metabolic network. Specifically, a few of the metabolic interactions predicted by MetQuest could be due to model artefacts. There may be other metabolic interactions happening between microbes, which could be absent in the metabolic network (gaps in the metabolic network). Due to this reason, such interactions may not be predicted by MetQuest. Further, our algorithm does not assign any weights to metabolites/paths. Although there are other methods like cFBA and OptCom, which make more quantitative predictions, they also demand better curated metabolic networks. Moreover, as with any graph-based technique, MetQuest also generates a large number of pathways between a given set of seed and target metabolites. All these pathways may not be active at a given time or in a given environment. We primarily see the use of MetQuest as a first-line investigatory tool, to cull down from a very large set of possible communities to a more tractable set, which may be studied further using EFM analyses, constraint-based techniques or through wet lab experiments.

In sum, we report a novel and efficient algorithm MetQuest, which (i) rapidly enumerates all possible biosynthetic pathways from a given metabolic network, (ii) efficiently handles branched and cyclic pathways, and most importantly, (iii) scales well to very large networks such as those representing microbial communities. Further, our algorithm requires only the genome-scale metabolic networks, a set of seed and target metabolites and a size cut-off, as against thermodynamic or atom-mapping information, thereby rendering it a very potent tool to perform large-scale analyses. Overall, we believe that MetQuest is a very useful tool for exploring and understanding metabolic pathways in large-scale networks, to generate testable hypotheses for further experiments.

References

1. Edwards, J. S. & Palsson, B. Ø. Systems properties of the *Haemophilus influenzae* Rd metabolic genotype. *J. Biol. Chem.* **274**, 17410–17416 (1999).
2. Reed, J. L., Vo, T. D., Schilling, C. H. & Palsson, B. Ø. An expanded genome-scale model of *Escherichia coli* K-12 (iJR904 GSM/GPR). *Genome Biol.* **4**, R54 (2003).
3. Oberhardt, M. A., Palsson, B. Ø. & Papin, J. A. Applications of genome-scale metabolic reconstructions. *Mol. Syst. Biol.* **5**, 320 (2009).
4. Monk, J., Nogales, J. & Palsson, B. Ø. Optimizing genome-scale network reconstructions. *Nat. Biotechnol.* **32**, 447–452 (2014).
5. Price, N. D., Reed, J. L. & Palsson, B. Ø. Genome-scale models of microbial cells: evaluating the consequences of constraints. *Nat. Rev. Microbiol.* **2**, 886–897 (2004).
6. Dräger, A. & Palsson, B. Ø. Improving collaboration by standardization efforts in systems biology. *Front. Bioeng. Biotechnol.* **2**, 1–20 (2014).
7. Ravikrishnan, A. & Raman, K. Critical assessment of genome-scale metabolic networks: the need for a unified standard. *Brief. Bioinform.* **16**, 1057–1068 (2015).
8. Handorf, T., Ebenhöh, O. & Heinrich, R. Expanding metabolic networks: Scopes of compounds, robustness, and evolution. *J. Mol. Evol.* **61**, 498–512 (2005).
9. Blum, T. & Kohlbacher, O. MetaRoute: Fast search for relevant metabolic routes for interactive network navigation and visualization. *Bioinformatics* **24**, 2108–2109 (2008).
10. Borenstein, E. & Feldman, M. W. Topological signatures of species interactions in metabolic networks. *J. Comput. Biol.* **16**, 191–200 (2009).
11. Lu, W., Tamura, T., Song, J. & Akutsu, T. Integer programming-based method for designing synthetic metabolic networks by Minimum Reaction Insertion in a Boolean model. *PLoS One* **9**, e92637 (2014).
12. Lu, W., Tamura, T., Song, J. & Akutsu, T. Computing smallest intervention strategies for multiple metabolic networks in a boolean model. *J. Comput. Biol.* **22**, 85–110 (2015).
13. Handorf, T. & Ebenhöh, O. MetaPath Online: A web server implementation of the network expansion algorithm. *Nucleic Acids Res.* **35**, 613–618 (2007).
14. Moriya, Y. *et al*. PathPred: An enzyme-catalyzed metabolic pathway prediction server. *Nucleic Acids Res.* **38**, W138–W143 (2010).
15. Karp, P. D. *et al*. Pathway Tools version 13.0: integrated software for pathway/genome informatics and systems biology. *Brief. Bioinform.* **11**, 40–79 (2010).
16. Pey, J., Prada, J., Beasley, J. E. & Planes, F. J. Path finding methods accounting for stoichiometry in metabolic networks. *Genome Biol.* **12**, R49 (2011).
17. Ma, H.-W. & Zeng, A.-P. Reconstruction of metabolic networks from genome data and analysis of their global structure for various organisms. *Bioinformatics* **19**, 270–277 (2003).
18. Mithani, A., Preston, G. M. & Hein, J. Rahnuma: Hypergraph-based tool for metabolic pathway prediction and network comparison. *Bioinformatics* **25**, 1831–1832 (2009).
19. Chou, C. H., Chang, W. C., Chiu, C. M., Huang, C. C. & Huang, H. D. FMM: A web server for metabolic pathway reconstruction and comparative analysis. *Nucleic Acids Res.* **37**, W129–W134 (2009).
20. McClymont, K. & Soyer, O. S. Metabolic tinker: an online tool for guiding the design of synthetic metabolic pathways. *Nucleic Acids Res.* **41**, e113 (2013).
21. Croes, D., Couche, F., Wodak, S. J. & van Helden, J. Metabolic PathFinding: inferring relevant pathways in biochemical networks. *Nucleic Acids Res.* **33**, W326–W330 (2005).
22. Croes, D., Couche, F., Wodak, S. J. & van Helden, J. Inferring meaningful pathways in weighted metabolic networks. *J. Mol. Biol.* **356**, 222–236 (2006).
23. Heath, A. P., Bennett, G. N. & Kavraki, L. E. An algorithm for efficient identification of branched metabolic pathways. *J. Comput. Biol.* **18**, 1575–1597 (2011).
24. Pitkänen, E., Jouhten, P. & Rousu, J. Inferring branching pathways in genome-scale metabolic networks. *BMC Syst. Biol.* **3**, 103 (2009).
25. Latendresse, M., Krummenacker, M. & Karp, P. D. Optimal metabolic route search based on atom mappings. *Bioinformatics* **30**, 2043–2050 (2014).
26. Schuster, S., Dandekar, T. & Fell, D. A. Detection of elementary flux modes in biochemical networks: a promising tool for pathway analysis and metabolic engineering. *Trends Biotechnol.* **17**, 53–60 (1999).
27. Schuster, S., Fell, D. A. & Dandekar, T. A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks. *Nature Biotechnol.* **18**, 326–332 (2000).
28. de Figueiredo, L. F. *et al*. Computing the shortest elementary flux modes in genome-scale metabolic networks. *Bioinformatics* **25**, 3158–3165 (2009).
29. Pérès, S., Felicori, L. & Molina, F. Elementary flux modes analysis of functional domain networks allows a better metabolic pathway interpretation. *PLoS One* **8**, e76143 (2013).
30. Vieira, G., Carnicer, M., Portais, J.-C. & Heux, S. FindPath: a Matlab solution for in silico design of synthetic metabolic pathways. *Bioinformatics* **30**, 2986–2988 (2014).

31. Van Klinken, J. B. & Willems Van Dijk, K. FluxModeCalculator: An efficient tool for large-scale flux mode computation. *Bioinformatics* **32**, 1265–1266 (2016).
32. Kamp, A. V. & Schuster, S. Metatool 5.0: fast and flexible elementary modes analysis. *Bioinformatics* **22**, 1930–1931 (2006).
33. Steffen, K. *et al.* Structural and functional analysis of cellular networks with CellNetAnalyzer. *BMC Syst. Biol.* **1**, 1–13 (2007).
34. Faust, K., Croes, D. & van Helden, J. In response to 'Can sugars be produced from fatty acids? A test case for pathway analysis tools'. *Bioinformatics* **25**, 3202–3205 (2009).
35. Stolyar, S. *et al.* Metabolic modeling of a mutualistic microbial community. *Mol. Syst. Biol.* **3**, 92 (2007).
36. Deville, Y., Gilbert, D., van Helden, J. & Wodak, S. J. An overview of data models for the analysis of biochemical pathways. *Brief. Bioinform.* **4**, 246–259 (2003).
37. Lee, C. Y. An algorithm for path connections and its applications. *IRE Transactions on Electronic Computers EC* **10**, 346–365 (1961).
38. Cormen, T. H., Leiserson, C. E., Rivest, R. L. & Stein, C. *Introduction to Algorithms* (The MIT Press, USA, 2009), 3rd edn.
39. Acuña, V. *et al.* Algorithms and complexity of enumerating minimal precursor sets in genome-wide metabolic networks. *Bioinformatics* **28**, 2474–2483 (2012).
40. Jaccard, P. Nouvelles recherches sur la distribution florale. *Bull. la Société Vaudoise des Sci. Nat.* **44**, 223–270 (1908).
41. Orth, J. D. *et al.* A comprehensive genome-scale reconstruction of *Escherichia coli* metabolism–2011. *Mol. Syst. Biol.* **7**, 535 (2011).
42. Jahreis, K., Pimentel-Schmitt, E. F., Brückner, R. & Titgemeyer, F. Ins and outs of glucose transport systems in eubacteria. *FEMS Microbiol. Rev.* **32**, 891–907 (2008).
43. Mahadevan, R. & Lovley, D. R. The degree of redundancy in metabolic genes is linked to mode of metabolism. *Biophys. J.* **94**, 1216–1220 (2008).
44. Court, S. J., Waclaw, B. & Allen, R. J. Lower glycolysis carries a higher flux than any biochemically possible alternative. *Nat. Commun.* **6**, 8427 (2015).
45. Mo, M. L., Palsson, B. Ø. & Herrgård, M. J. Connecting extracellular metabolomic measurements to intracellular flux states in yeast. *BMC Syst. Biol.* **3**, 37 (2009).
46. Nogales, J., Palsson, B. Ø. & Thiele, I. A genome-scale metabolic reconstruction of *Pseudomonas putida* KT2440: iJN746 as a cell factory. *BMC Syst. Biol.* **2**, 79 (2008).
47. Kumar, A., Kumar, S. & Kumar, S. Biodegradation kinetics of phenol and catechol using *Pseudomonas putida* MTCC 1194. *Biochem. Engg. J.* **22**, 151–159 (2005).
48. Feist, C. F. & Hegeman, G. D. Phenol and benzoate metabolism by *Pseudomonas putida*: regulation of tangential pathways. *J. Bacteriol.* **100**, 869–877 (1969).
49. Hadadi, N., Hafner, J., Shajkofci, A., Zisaki, A. & Hatzimanikatis, V. ATLAS of Biochemistry: A repository of all possible biochemical reactions for synthetic biology and metabolic engineering studies. *ACS Synth. Biol.* **5**, 1155–1166 (2016).
50. Kanehisa, M., Sato, Y., Kawashima, M., Furumichi, M. & Tanabe, M. KEGG as a reference resource for gene and protein annotation. *Nucleic Acids Res.* **44**, D457–D462 (2016).
51. Martin, V. J. J., Pitera, D. J., Withers, S. T., Newman, J. D. & Keasling, J. D. Engineering a mevalonate pathway in *Escherichia coli* for production of terpenoids. *Nature Biotechnol.* **21**, 796–802 (2003).
52. Faust, K. & Raes, J. Microbial interactions: from networks to models. *Nat. Rev. Microbiol.* **10**, 538–550 (2012).
53. Zomorrodi, A. R. & Maranas, C. D. OptCom: a multi-level optimization framework for the metabolic modeling and analysis of microbial communities. *PLoS Comput. Biol.* **8**, e1002363 (2012).
54. Khandelwal, R. A., Olivier, B. G., Röling, W. F. M., Teusink, B. & Bruggeman, F. J. Community flux balance analysis for microbial consortia at balanced growth. *PLoS One* **8**, e64567 (2013).
55. Tzamali, E., Poirazi, P., Tollis, I. G. & Reczko, M. A computational exploration of bacterial metabolic diversity identifying metabolic interactions and growth-efficient strain communities. *BMC Syst. Biol.* **5**, 167 (2011).
56. Miller, L. D. *et al.* Establishment and metabolic analysis of a model microbial community for understanding trophic and electron accepting interactions of subsurface anaerobic environments. *BMC Microbiol.* **10**, 149 (2010).
57. Büchel, F. *et al.* Path2Models: large-scale generation of computational models from biochemical pathway maps. *BMC Syst. Biol.* **7**, 116 (2013).
58. Orth, J. D., Palsson, B. Ø. & Fleming, R. M. T. Reconstruction and use of microbial metabolic networks: the core *escherichia coli* metabolic model as an educational guide. *EcoSal Plus* **4** (2010).
59. Flahaut, N. A. L. *et al.* Genome-scale metabolic model for *Lactococcus lactis* MG1363 and its application to the analysis of flavor formation. *Appl. Microbiol. Biotechnol.* **97**, 8729–8739 (2013).
60. Balagurunathan, B., Jonnalagadda, S., Tan, L. & Srinivasan, R. Reconstruction and analysis of a genome-scale metabolic model for *Scheffersomyces stipitis*. *Microb. Cell Fact.* **11**, 27 (2012).

Acknowledgements

This work was supported by the Indian Institute of Technology Madras grant ERP/1314/004/RESF/KARH to K.R. and the INSPIRE fellowship, Department of Science and Technology, Government of India to A.R.

Author Contributions

K.R. conceived the study. A.R., K.R. and M.N. devised the methodology. A.R. implemented the algorithm and prepared the original draft of the manuscript. All authors revised, read and approved the final manuscript.

Additional Information

Supplementary information accompanies this paper at https://doi.org/10.1038/s41598-018-28007-7.

Competing Interests: The authors declare no competing interests.

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