University of Geneva

Master thesis

Metabolic network analysis: Towards the construction of a meaningful network

Author

Nikolaus Fortelny

Nikolaus.Fortelny@isb-sib.ch

Supervisor

Frederique Lisacek

August 25, 2010

Contents

1 Abstract

Investigations of cell metabolism were originally done through the identification of metabolic pathways with biochemical methods. With the extended use of bioinformatics, more systematic approaches allow examining whole metabolic networks or reconstructing metabolic networks from genomic data. Graph theory is the most common method used for the analysis of networks. However, uncertainty exists on how to apply graph theory to metabolic networks, more precisely on how to represent existing data and what conclusions can be drawn from the representation. In this work, we first assessed the structural and topological consequences of using data derived from different sources and curated with different methods. From this study we confirmed that the treatment of currency metabolites is the main influence on network structure. Then we created a network which, in our opinion, enhances the representation of biological information and investigated network failure through node removal. We found that, in a bipartite network, removing compound nodes with respect to their betweenness breaks down the network the fastest. We concluded that high betweenness compounds should therefore be considered as connectors between the sub-networks in the metabolic network. Furthermore, we reinforced arguments showing that the structure of a metabolic network does not maximize resistance to directed node removal (network attack) compared to standard random networks, as thought intuitively and put forward by other authors.

2 Introduction

Understanding the different parts of an organism that make up a living being is a very challenging task for humans. We therefore, in order to simplify, like to separate the task into smaller sub-tasks which can then be understood on their own. One of these steps towards understanding how entire organisms function involves understanding metabolism. Essentially, metabolism is seen as an open system: Nutrients are taken up by the cell and are then consumed to produce energy and to build up proteins, DNA and other molecules which are needed to keep the cell working and to proliferate. Waste products and energy is released into the environment. To understand the way these processes work we have to first understand that the consumption and production of the metabolites cannot be organized arbitrarily but is, in fact, rather tightly regulated. Through enzymes a cell increases or decreases the number of molecules converted specifically for each reaction. In a similar way the uptake and release of compounds in the environment is regulated. From this description we can pose a series of questions : What inputs can an organism use, what output can it produce from these and how efficient and fast is this conversion? From the answers of these questions we will then be able to explain the workings of one specific cell type but also the differences between different cells. A common strategy for addressing these questions is to identify metabolic pathways, i.e. a series of chemical reactions which convert nutrients from the environment into cell components. Once we know the set of all existing pathways in a given organism we conclude how the metabolism operates.

Historically, pathway identification has been done on the basis of biochemically determined mutant phenotypes. If a gene is mutated, the corresponding enzyme is not available any more and the catalyzed reaction might not happen quickly enough. Consequently, all products which depend on this reaction will not be produced any more, so the cell will need the given product in its environment to survive. This is called auxotrophy phenotype (e. g.: auxotrophy for a given amino acid). By mutating all genes and comparing the phenotypes, enzymes were then grouped in clusters. These clusters are called pathways because they usually represent series of reactions leading from one compound to the other. However these pathways were identified only in some model organisms and the organization of the metabolic network can change considerably between organisms depending on the active enzymes.

More recently, with the huge amount of genomic data collected today in-silico [14, chapter 10] approaches were developed to assist the analysis of reconstructed networks with less resources. One popular approach entails modeling a metabolic network by using graph theory and exploring paths in the model. It is therefore called metabolic path finding. For example (for other possibilities see section 2.2) a metabolic network can be represented as nodes (compounds) which are connected by a link (reaction) if they are interconvertible by this reaction. Apart from being able to analyze a lot more data, metabolic path finding can also be used to find pathways which are more tricky or even impossible to identify in biochemical experiments (e. g.: pathways with many alternatives or lethal phenotypes). However this approach is hampered by some serious

problems: Depending on the quality of the model there can be a lot of possible pathways found, many of which can be suspected to be false positives, meaning that they are biochemically irrelevant (explained in [24]). When data is not complete some pathways will not be identified either (false negatives). Because there is no organism whose metabolism is understood (i. e. for which the entire set of biochemically meaningful pathways is known) both, false negatives and positives, can only be estimated. No simple test of the reliability of pathway identification approaches is available yet.

Because there is no easy validation possible, great care must be taken to best represent the biological data when modeling a metabolic network. There are some issues to be pointed out:

- The biological data must be well understood.
- A meaningful representation must be defined.
- Complete and correct biological data input must be used.
- One special feature of metabolic networks is the bias of path finding caused by currency metabolites. These metabolites are found in many reactions and lead to meaningless shortcuts in the network.

Once a meaningful network is created, it can be used for other studies as well: One example is the study of networks resistance to perturbations. In such a study one tries to break down a network with different methods in order to identify essential enzymes of a metabolism [19] or to make a statement about general design principles of robust networks [1]. However we first have to understand the data we are dealing with.

2.1 Nature of data

Basically a metabolic network is composed of reactions which connect the metabolites. Reactions have been extensively studied in the discipline of physical chemistry. A reaction can be seen as two reactions, one going in one direction (converting substrates to products) and one in the other (converting products to substrates). Because one direction can be energetically favorable (figure 1) to the other, molecules get converted in this direction faster. Since the reactions in the two inverse directions happen at the same time and one is faster than the other, the reaction will reach an equilibrium at some point where the amount of molecules converted from substrate to product is the same as the reverse. This fixed ratio of concentrations is quantified by the equilibrium constant K, which is constant for a given temperature and pressure. When the ratio of concentrations is not corresponding to the equilibrium constant, we observe the reaction to move in one direction, which is towards the equilibrium.

Enzymes are protein catalysts which increase the velocity of a reaction so it will reach the equilibrium faster (figure 1). Some reactions might be too slow to be observed under normal influences but with the help of enzymes they occur at a significant rate in the metabolism. Enzymes do not, however, change the equilibrium constant because they

influence the reactions in both directions equally. Unfortunately for modeling purposes, enzymes are a very variable factor. They can change in concentration and activity in very short time, which has a great influence on the performance of the enzyme and therefore on the velocity of the reaction. Examples of factors, which influence the activity of enzymes, are post-translational modifications, substrate and product concentrations and concentrations of other compounds in the same pathway.

Finally, to analyze the entire metabolism is to see it as a whole network of such reactions, each with different velocity, catalyzing enzymes and equilibrium constant. This means that substrates and products of one reaction also take part in other reactions. As a consequence, some reactions of a network never come to the equilibrium state because their products are permanently removed. Consequently these reactions are observed in one net direction only which is towards the equilibrium state. In contrast to reversible reactions which were observed to go both ways these reactions were called irreversible.

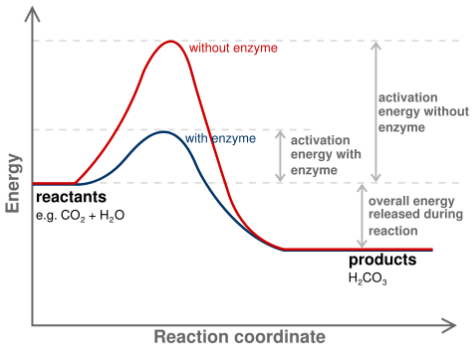


Figure 1: An enzyme's effect on a reaction.

2.2 Representation

Representing metabolism is not an obvious thing to do. Amongst current methods examples are graph theory, flux balance analysis [6] or petri nets [4].

When analyzing data with graph theory there are some choices to make. These will predefine the meaning of measurements taken from the graph. Typically when dealing with metabolism we can use compounds as nodes and reactions as links (compound graph) or the reverse (reaction graph). We can also use a bipartite graph with reactions and compounds as nodes (see [14, p. 234], [26] and references therein). Other representations have been developed, mainly to tackle the problem of currency metabolites (described in section 2.4).

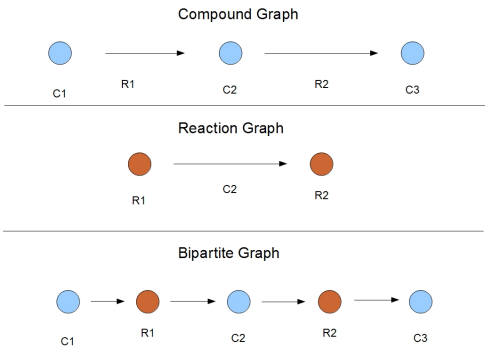


Figure 2: The usually used ways of representing metabolism by graph theory.

Apart from choosing the type of arcs and nodes of the network we have the choice between weighted or unweighted, directed or undirected graphs, as well as the possibility

to use hyper- and sparse-graphs. Which ones of these options we choose depends on the data we represent and the conclusions we want to draw .

2.3 Making a graph model

In order to analyze our network of reactions, the metabolism, with one of the above graph models we start with one approximation: The final network contains links between metabolites A and B if and only if A can be converted into B at given conditions (temperature, pressure, pH) at considerable velocity. Because all possible reactions happen at some rate (which can be very low), this is like choosing a cutoff for the velocity of a reaction. If the velocity of the reaction is below the cutoff, it is not allowed in the network.

In practice, finding all reactions which occur at considerable speed is done by mapping EC (Enzyme Commission numbers) annotation of genes to the corresponding reactions found in a database of metabolic reactions [14, p. 236] (e. g. KEGG (http://kegg.jp/, [16, 17, 15]) , RhEA (www.ebi.ac.uk/rhea/) or MetaCyc (http://metacyc.org/, [3])). So we approximate some more by stating that all the annotated enzymes are expressed and they catalyze a reaction at a significant velocity [19]. This is clearly against the current belief in biology which says that organisms turn enzymes on and off under different environmental conditions, compartments and time points depending on their need.

Apart from the above, there are some technical issues to be considered as well:

In addition to errors caused by possible annotation mistakes and missing information, some metabolic databases contain generic compounds (without a concrete chemical formula) and multiple entries for metabolites (e. g.: glycans in KEGG). These should not be incorporated directly into a network where metabolites should have a well defined formula. Correction of these details can be laborious but it is non-ambiguous.

There are some more difficult tasks to be addressed: We need to add non-catalyzed reactions, to precisely link enzymes (their protein or gene sequences) to catalyzed reactions and to see how we can incorporate reversibility information in the simulated network.

By non-catalyzed reactions we mean reactions, which occur without a catalyst at considerable speed in an organism. If we simply connect EC annotations of genes to reactions, the non-catalyzed ones will be neglected (pointed out by [5]). To correct this neglect, non-catalyzed reactions can be added if they can be identified.

Even without considering this problem, inferring existing reactions from EC annotation alone can be erroneous. There are two steps to the procedure of linking genes to reactions: First EC annotation has to be done according to the DNA or protein sequence and second the EC annotation has to be linked to reactions catalyzed. Both of these steps are done with uncertainties. Gene annotation is not always done with certainty and EC numbers are not always specific enough to determine the concrete reaction(s) really catalyzed by this enzyme. Multiple enzymes can perform the same reaction and

multiple reactions can be performed by a single enzyme. The link between EC numbers and their catalyzed reactions is not obvious either: Some EC numbers can be unspecific and apply to different substances or others describe reactions which could actually be broken down into many smaller ones. The way these links are treated explains the differences between metabolic databases. For example, the reaction associated with EC 4.2.1.20 (Tryptophan synthase, reaction (1)) can be broken down into two reactions (reactions (2) and (3)). In KEGG we can find all three of these reactions while in Rhea only (1) is found and MetaCyc has (1) and (3) stored.

$$L\text{-}Ser + Indoleglycerol\text{-}Phos \leftrightarrow L\text{-}Trp + D\text{-}Glycerald\text{-}3\text{-}Phos + H_2O \quad (1)$$

$$Indoleglycerol\text{-}Phos \leftrightarrow Indole + D\text{-}Glycerald\text{-}3\text{-}Phos \quad (2)$$

$$L\text{-}Ser + Indole \leftrightarrow L\text{-}Trp + H_2O \quad (3)$$

Another issue concerning the data used in the simulation, which can have an impact on the network structure, is the reversibility information of reactions. Every reaction is theoretically reversible as explained earlier (section 2.1) [12]. Since some reactions are observed in permanent disequilibrium they were seen as irreversible because the net flux was permanently going in one direction. This information can change, depending for example on the context it is found in . Looking at KEGG data for instance, the same reaction can be reversible or irreversible in different pathway files. Even though this question seems quite delicate, if a reaction is permanently observed to work in one direction only, it might be useful to consider it as an irreversible reaction. If reversibility information is considered, each reaction is usually split into two, reaction and reverse reaction, and a directed graph is used.

2.4 Currency metabolites

Even if all the obstacles mentioned above were overcome, say the input data is correct and complete and the optimal representation is used, care must be taken to avoid the problem of currency or pool metabolites mentioned by many authors [12, 21, 14].

The problem of currency metabolites can be best illustrated with an example:

Consider two reactions inside a network:

$$Methanol + H_2O_2 \leftrightarrow Formaldehyde + 2H_2O \tag{4}$$

$$N^6\text{-}Acetyl\text{-}L\text{-}Lysine + H_2O \leftrightarrow Acetate + L\text{-}Lysine \tag{5}$$

Now suppose for both reaction all substrates are connected to all products. We will find the path ($Methanol \rightarrow H_20 \rightarrow L\text{-}Lysine$). L-Lysine can certainly not be produced from water alone. So here, water leads to a chemically meaningless shortcut and the path found is nonsense. Water and other molecules (ATP, ADP, NAD, NADP, H^+,...) are called currency metabolites. They take part in many reactions in the cell thereby creating many meaningless shortcuts like the one explained above hence any path-based measurement of the graph will be biased. This led to the probably wrong conclusion that the average path in many metabolic networks is about 3 which is now believed to be much higher [7, 2]. There is no consensus on how to treat this problem.

Many authors have tried to circumvent it with different methods (see also [7] and references therein]):

2.4.1 Penalizing metabolites

Early efforts include simply removing some compounds believed to be currency metabolites because of biochemical knowledge. Later systematic measures like node degree were used to define compounds to be removed [12, 8]. Similarly node degree was used as a penalty weight for metabolites in lightest path finding [5]. Lightest path finding means finding the path which minimizes the sum of weights assigned to the links of this path.

Figures 3 to 5 show the ambiguity when trying to simply delete currency metabolites. We can see that when some compounds (H_20, H^+, NAD, NADH) (which are currency metabolites in most reactions) are removed the network is still intact but less tightly connected than before. If we delete β-*Alanine* from the network it will be fragmented in two. In this network β-*Alanine* has a high degree so, if we use node degree to characterize currency metabolites, it is identified as one of them. However we can expect that this is not true for all reactions but can be true for some of them.

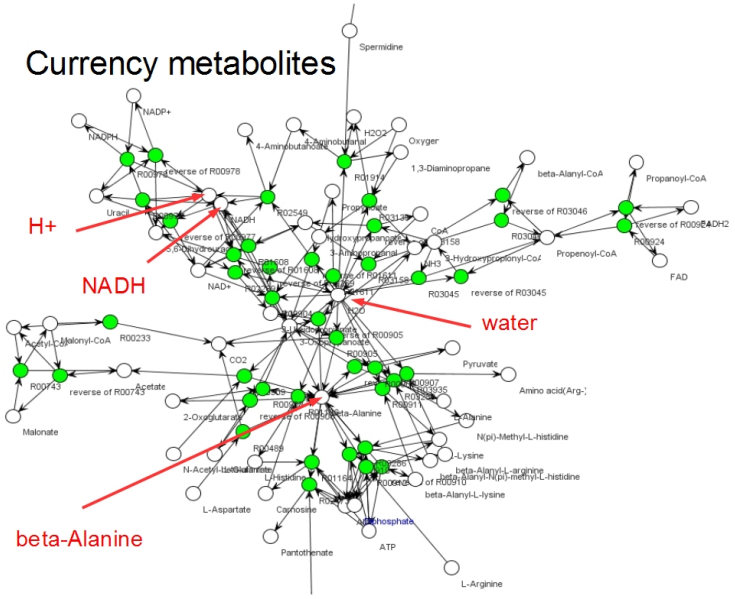


Figure 3: A partial metabolic network, some highly connected compounds are highlighted. A lot of paths pass by those nodes but many of them have no biochemical relevance.

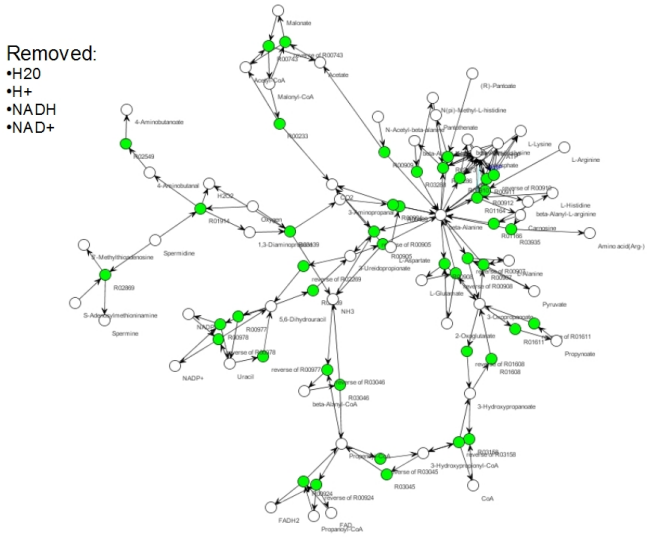


Figure 4: The compounds usually identified as currency metabolites are removed from the graph (H_20, H^+, NAD, NADH).

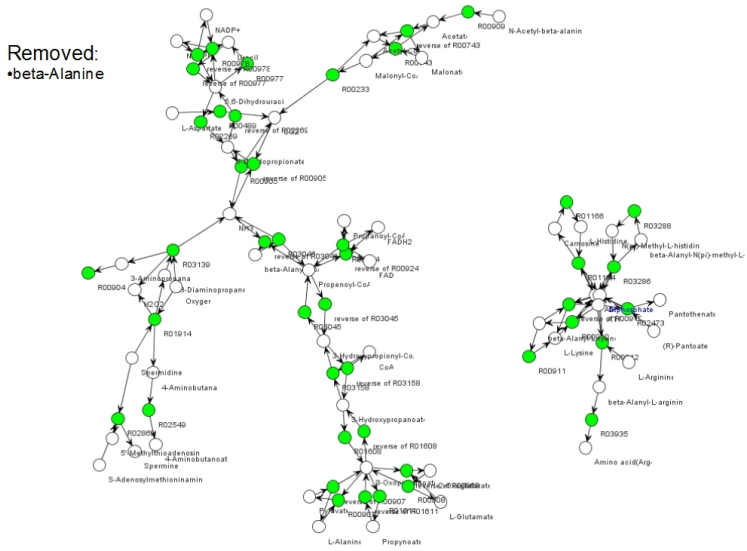


Figure 5: Beta-Alanine, a compound with a high in-degree was also removed, however it is usually not considered to be a currency metabolite. This splits the graph in two. This example shows the difficulties when trying to tackle the problem of currency metabolites.

2.4.2 Integrating chemical knowledge

Another way to tackle the problem of currency metabolites was introduced by tracing atoms of molecules in the network [2]. In such a network, shortcuts for metabolite conversion are avoided. Since in this case atoms are traced in the network, paths of molecules cannot be found any more, only paths of atoms. So it was not used to identify biochemical pathways (of molecules) but to analyze the average path length of the network.

A related approach is the RPAIR information [18] in the KEGG LIGAND database. There, reactant pairs (one substrate and one product) are classified in five groups: main, cofactor, transferring groups, nucleotide ligase and separation / addition of inorganic compounds. Evidently this classification is based on the role of the two reactants of one reaction which relates to how the metabolites are changed chemically. Currently all reactions in the KEGG database have their reactants assigned to one of these five classes. Exploiting this information, it is possible to construct a graph where the main metabolites are defined for every reaction. Compared to older graph representations, path finding in such graphs showed to have an increased probability of identifying annotated pathways [7].

2.4.3 Global versus local definition

Very early it was noticed by [21] that defining currency metabolites for the whole network was misleading. They therefore constructed a database by manually defining and unlinking interfering metabolites for each reactions separately. The classification in RPAIR extends this approach. Instead of defining currency metabolites globally (for the whole network) by removing or penalizing them, they are defined locally, i. e. for each reaction separately. This means that the same metabolite can be a main metabolite in one reaction and a cofactor in another.

There is an important difference between these two approaches: It was observed that a global definition of currency metabolites prevents path-finding algorithms from detecting paths where those compounds are primary metabolites [5].

The problem can be illustrated with the following examples (see also [21] for further explanation):

$$ATP + D\text{-}Glucose\text{-}1\text{-}phosphate \leftrightarrow Diphosphate + ADP\text{-}glucose \tag{6}$$

$$ATP + Pyridoxal \leftrightarrow ADP + Pyridoxalphosphate \tag{7}$$

$$ATP + H_2O \leftrightarrow ADP + Orthophosphate \tag{8}$$

$$ATP + AMP \leftrightarrow 2\,ADP \tag{9}$$

In reactions 6 and 7 the main pathway can still be found ($D\text{-}Glucose1\text{-}phosphate \rightarrow ADP\text{-}glucose, Pyridoxal \rightarrow Pyridoxalphosphate$) when H_20, Orthophosphate, ADP and ATP is removed. Reactions 8 and 9 just don't happen. Since ATP and ADP are not present in the network the paths for their synthesis cannot be found.

So when is this problematic? The problem arises if currency metabolites are not purely used as cofactors in the network meaning there is a pathway using the metabolite to produce some other metabolite. An example is the metabolic pathway producing 1-Methyl-4-pyridone-3-carboximide (KEGG ID: C05843) from NAD^+ (KEGG PATH: ko00760). If NAD^+ is deleted from the network (and some reactions are not considered to be reversible as it is the case in this pathway) 1-Methyl-4-pyridone-3-carboximide cannot be produced from the network any more.

Finally note that currency metabolites (or external compounds) are also defined in other types of investigations like scopes of compounds [10] or flux balance analysis [24].

2.5 Dataset comparison

As already pointed out validation of path finding experiments is problematic. To be able to see the effect of the issues pointed out in sections 2.2, 2.3 and 2.4 we compare networks from different databases (section 3.1.1), without the use of generic compounds and glycans (section 3.1.2) and with the use of RPAIR data (section 3.1.3).

For all analysis we created a bipartite, undirected and unweighted network with nodes for reactions and compounds. For each reaction we have one node which is connected to all the compounds taking part in this reaction. A bipartite network was used to enable us to look at reaction and compounds nodes separately. Directed edges could have been included if some reactions were considered irreversible. Some sources contain such information but we chose to use an undirected network because the information was not clear and all reactions are reversible in theory. Furthermore it was observed [7] that not including direction is not drastically changing the results.

To compare the different sets the following measurements were taken (for further explanations see also [11, 22] and methods, section 6.3, for details) :

Note that in a bipartite network, like the metabolic network here, any measure, which applies to nodes, can be taken from reactions, compounds or both. The **Cluster coefficient distribution** quantifies the inter-connectivity of nodes. It is defined as the fraction of existing links amongst all links possible between one node's neighbors. If it takes the value 1 all neighbors are connected, if it takes the value 0 there are no connections between the neighbors. So it informs about how tightly connected the nodes are in the network. In order to calculate the cluster coefficient one has to decide to analyze either the compound or reaction network. This is because in the bipartite network neighbors of nodes are never connected (neighbors of reactions are always compounds which will never be connected and vice verse). So the measure would always be 0 in a metabolic, bipartite network.

The **cumulative degree distribution** [20] of the network tells us the probability of a node having a degree higher or equal to degree x. The degree of a node in an undirected graph is the number of arcs it is an endpoint of. In a directed graph one usually differentiates between the "incoming" and "outgoing" degree. To show, that the degree distribution of metabolic networks follow a power-law distribution, was used to argue that they are scale-free networks [13]. This has been questioned later and is probably an ongoing discussion [20]. Even if not used to make a statement about scale-freeness the distribution still tells us about the connectivity of the nodes in the network and to compare different networks.

Measuring and plotting the **correlation between cluster coefficient and degree** was used to make a statement about the modularity of the network: It was observed that in many empirical network this correlation is negative meaning that highly connected nodes have a low clustering coefficient and vice verse [23]. From this observation it was concluded that the highly connected nodes are connectors between clusters and that these clusters are made up out of less connected nodes which are highly interconnected

with their neighbors. Again (like for cluster coefficients themselves) we have to decide to use a reaction or compound network.

The measures of **tolerance to network attack and error** were introduced by [1]. Both are based on the observation of the effect of node removal on a given network. In the case of error randomly selected nodes are removed while in the case of attack important nodes are removed first.

There are many measures of importance (or centrality) [11, 22, 14] of nodes in a network such as degree, betweenness and closeness. The degree measure is explained above. To calculate node betweenness we count the number of shortest paths in the network a given node is part of. The entire set of shortest paths in a network denotes for each possible pair of nodes the shortest path connecting them. Finally closeness is a measure which is high if the given node is connected to most nodes in few steps and to few nodes in many steps. These values can be calculated once at the beginning and then the initial value can be reused every time or it is recalculated after each node removal.

When nodes are removed from the network a measure is needed to analyze the effect of the removal on the network. Again we have different possibilities: The diameter of a network is the longest of all shortest paths. It tells us about how easy it is to maneuver in the network. When nodes are removed paths are disrupted and the diameter of the network increases. The largest connected component is the highest number of nodes which can be found to be connected. It always decreases slowly when nodes are removed because the size of the network shrinks with each deleted node. When parts of the network become disconnected from each other, the measure decreases more rapidly.

After having defined the necessary measurements, we can start our investigation and study real network data.

3 Results

3.1 Input datasets

The first step in our analysis was to use different datasets and curation methods to compare the networks created from each of these. The effect of the different methods to tackle issues described in the introduction (section 2.3 and 2.4) can thereby be assessed.

E. coli was used as a model organism in all datasets. We assume that this model allows us to construct the most complete network possible. As an alternative we could have integrated reactions from multiple organisms into one joint network. However such a network does supposedly not exist in biological reality, so the conclusions drawn from it might not apply to real networks.

3.1.1 Four datasets

First we compared metabolic networks constructed from three different databases. These are KEGG, RhEA and MetaCyc.

Three datasets **KEGG EC**, **RhEA EC** and **MetaCyc EC** were produced by first collecting all EC annotations of the E.coli proteome taken from UniProtKB/Swiss-Prot (http://www.expasy.org/sprot/) and then retrieving the reactions linked to the EC numbers. This link between EC number and reaction can be different in the databases.

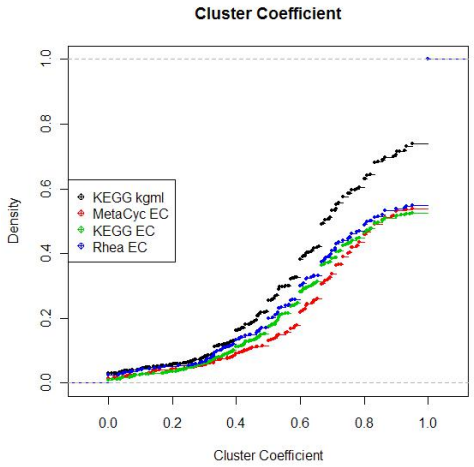
The **KEGG kgml** dataset consists of all the reactions found in the "eco01100.kgml" file from KEGG. The precise reaction equation was taken from the "reaction.lst" file of KEGG LIGAND because it is not complete in the kgml file. Remark that KEGG ".kgml" files contain all reactions suspected to occur in one pathway, the eco01100.kgml contains all reactions thought to occur in E. coli (contains all reactions from the KEGG pathway "ko01100 Metabolic pathways"). Therefore the difference between the datasets KEGG kgml and KEGG EC should be the presence of uncatalyzed reactions. We expect to find them in KEGG kgml but not in KEGG EC.

	KEGG kgml	KEGG EC	RhEA EC	MetaCyc EC
Size (nodes)	3067	2672	1278	1398
Reaction nodes	1477	1327	615	666
Compound nodes	1590	1345	663	732
Edges	6177	5549	2651	2983

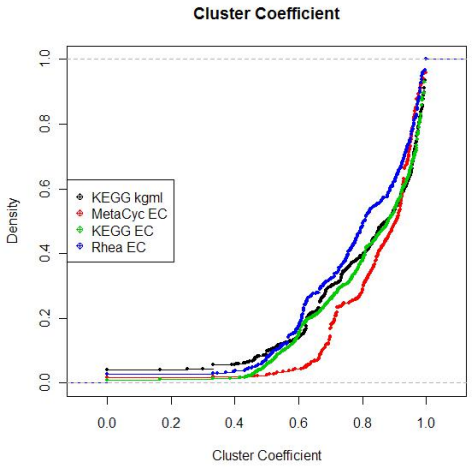
Table 1: Number of the nodes and edges in the different networks.

The comparison of the four datasets using the different measures applied does not lead to a clear conclusion. The differences between the datasets seem to be quite small:

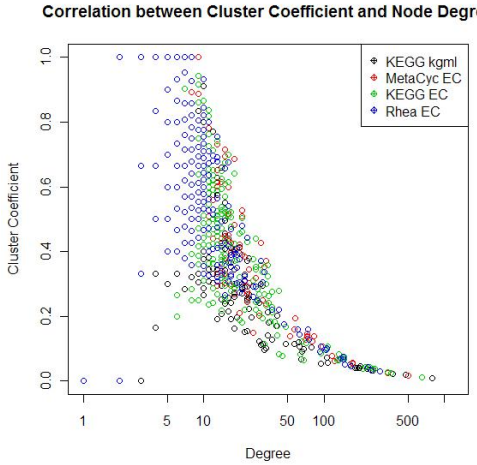
Figures 6(a) and 6(b) show the cluster coefficients plotted against their density function (as explained in section 6.3). There seem to be many nodes with a very high cluster



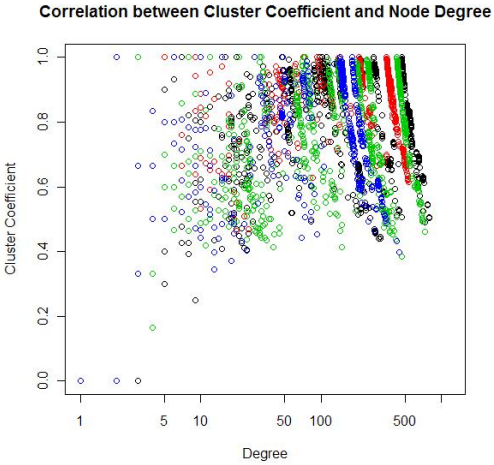
(a) cluster coefficient in the compound graph



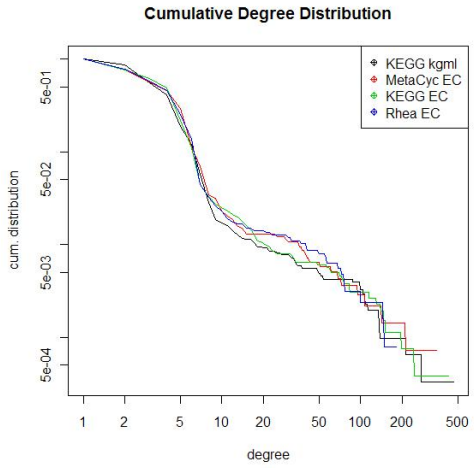
(b) cluster coefficient in the reaction graph



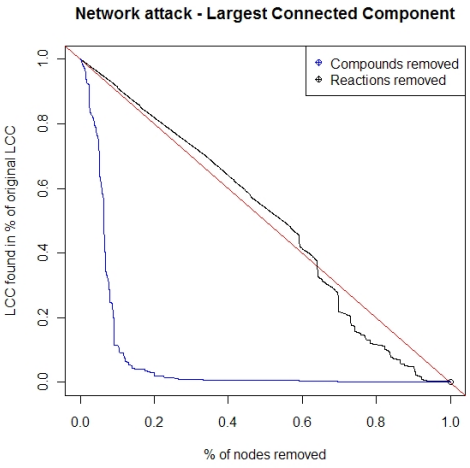
(c) cluster coefficient plotted against node degree in the compound graph



(d) cluster coefficient plotted against node degree in the reaction graph



(e) cumulative degree distribution



(f) network attack KEGG kgml

Figure 6: Different measurements of the four data sets

	Correlation coefficient	p-value
Compound Graph		
KEGG kgml	-0.284	7.886e-31
MetaCyc EC	-0.431	3.689e-34
KEGG EC	-0.396	1.297e-50
RHEA EC	-0.440	2.110e-32
Reaction Graph		
KEGG kgml	0.143	3.272e-08
MetaCyc EC	0.106	0.006
KEGG EC	0.161	4.559e-09
RHEA EC	0.072	0.076

Table 2: correlation coefficients between node degree and cluster coefficients

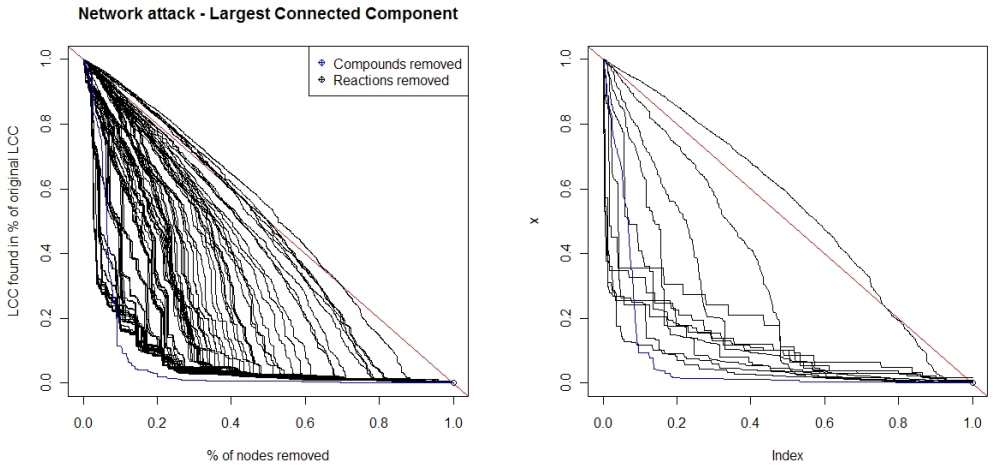
coefficient in both graphs (reaction graph and compound graph). This means that in all datasets the nodes are tightly connected. Recall that these graphs have not been treated to avoid the problem of currency metabolites.

From figure 6(c) and table 2 we can see a significant correlation between cluster coefficient and node degree in the compound graph for all datasets. Figure 6(d) and table 2 show that this correlation is much weaker in the reaction network.

Figure 6(e) shows the cumulative degree distribution of the 4 datasets. The functions are very similar. However, from table 1 we can see that the datasets clearly differ in size. RhEA and MetaCyc seem to assign less reactions to the same EC numbers than KEGG. The difference between KEGG EC and KEGG kgml probably shows the amount of non-catalyzed reactions in the network as explained in the introduction (section 2.3). We can further see that in all networks we have the same ratio of nodes to edges which is 0,5.

From all of these measurements together we can see that the networks created have very similar features even though they differ strongly in size.

One remarkable feature of all the networks (shown for KEGG kgml and MetaCyc) is that (in their raw form) they show a higher resistance to the removal of reaction nodes than to compound nodes, when both types are removed stepwise and ordered by their degree (figure 6(f)). Note that this measure is also highly influenced by currency metabolites mentioned earlier (section 2.4). To tackle the problem we therefore removed step by step the compounds with the highest degree [12] before analyzing the network vulnerability to removal of reaction nodes again. Even if we think that removal of highly connected nodes is not the best way to tackle the problem of currency metabolites we can still observe how many compound nodes have to be removed to generate a similar curve for compounds and reactions. Results are shown for the removal of 0 to 100 metabolites (figure 7(a)) prior to removal of all reaction nodes. So we observe that only after the removal of roughly 50 metabolites the two curves (compounds and reaction removal)



(a) Network attack on the KEGG kgml network with stepwise removed compounds. From right to left: Metabolites with high degrees are removed one by one so every line represents a metabolic graph with one additional metabolite removed compared to the previous line / graph.

(b) Network attack on the MetaCyc network with stepwise removed compounds. From left to right: The graph was treated the same way as for the KEGG kgml only the metabolites were removed in steps of 10 instead of 1.

Figure 7: Network attack with removed "currency" metabolites

get close to one another. We observe a similar feature for the MetaCyc network (figure 7(b)). However, we do not expect to find 50 currency metabolites in the metabolism (the number identified usually ranges between 10 and 20 in the articles introduced in section 2.4). So we conclude that the network is more tolerant to the removal of reaction rather than compound nodes.

3.1.2 Curation methods

Different ways of treating the data derived from the KEGG kgml dataset (section 3.1.1) were compared to evaluate the effect of those curation methods. Raw data was compared to data with removed glycans and/or generic metabolites.

No glycan data was retrieved the same way as the KEGG kgml data but without using glycans. Glycans meant here are those substrates or products whose ID is G plus five letters (compared to compounds which is C plus five letters). Some glycans also have references to compound IDs so it is possible that, after deleting the glycan entry, the molecules are still found in the graph because of their compound ID.

No generic data was retrieved like the KEGG kgml but by excluding all compounds that had an 'n' or an 'R' in their chemical formula stored in KEGG LIGAND. 'n' corresponds to an unspecified number of a specific chemical groups and 'R' depicts a unspecified chemical group. Usually if we find a generic compound in a reaction in KEGG (equation 10) we find the same reaction with the concrete compounds as well (equations 11 and 12). So if we just delete the generic compounds the specified reactions stay which is the goal here. However if not all concrete reactions exist already, we will miss those.

$$NTP + Pyruvate \leftrightarrow NDP + Phosphoenolpyruvate \tag{10}$$

$$ATP + Pyruvate \leftrightarrow ADP + Phosphoenolpyruvate \tag{11}$$

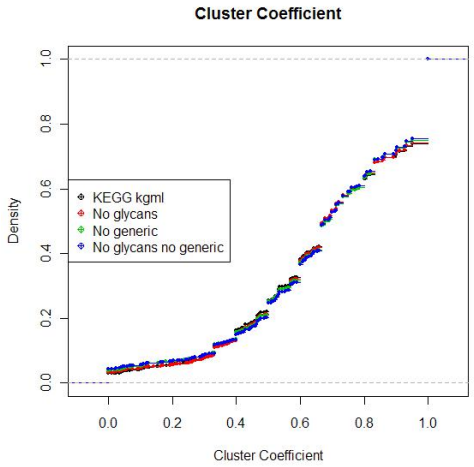
$$GTP + Pyruvate \leftrightarrow GDP + Phosphoenolpyruvate \tag{12}$$

No Glycan No Generic doesn't include neither glycans nor generic molecules as described above.

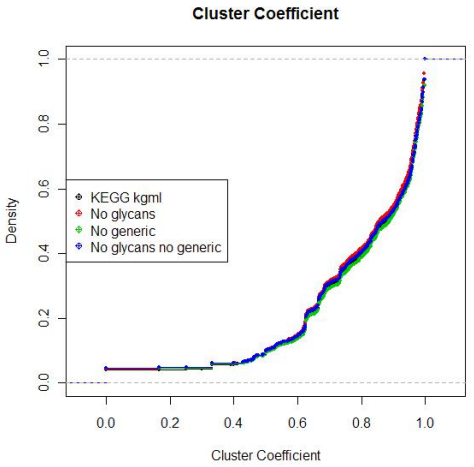
	KEGG kgml	No glycans	No generic	No glycans No generic
Size (nodes)	3067	2940	2914	2790
Reaction nodes	1477	1477	1477	1477
Compound nodes	1590	1463	1437	1313
Edges	6177	5777	5852	5452

Table 3: Number of the nodes and edges in the different networks.

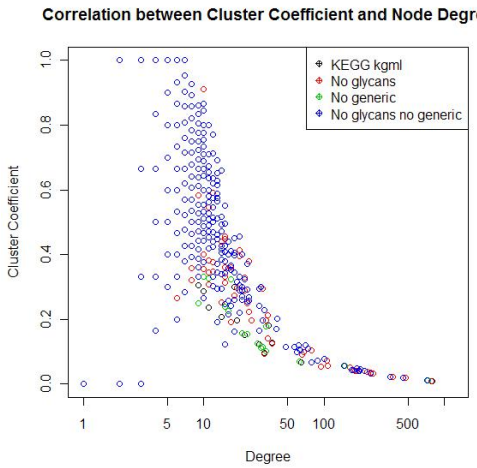
None of the measurements (described in section 2.5) showed any substantial differences between the datasets. Looking at the plots (figure 8) and table 4 we can see that the data is almost exactly the same. When removing glycans and generic metabolites the networks size does not change a lot either (table 3). This indicates limited structural consequences of curation methods.



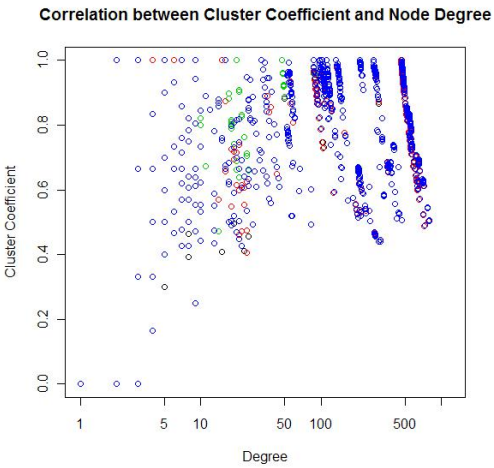
(a) cluster coefficient in the compound graph



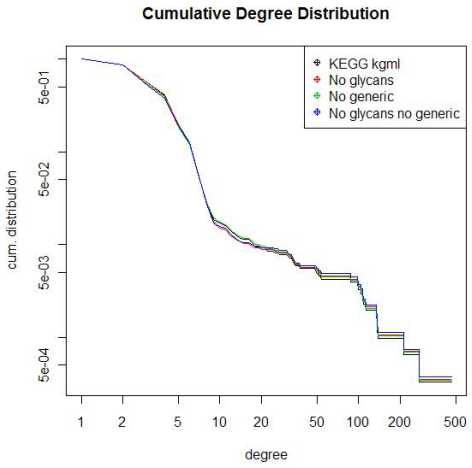
(b) cluster coefficient in the reaction graph



(c) cluster coefficient plotted against node degree in the compound graph



(d) cluster coefficient plotted against node degree in the reaction graph



(e) cumulative degree distribution

Figure 8: Different measurements of the curated data sets

	Correlation coefficient	p-value
Compound Graph		
KEGG kgml	-0.28	7.886e-31
No glycans	-0.28	2.901e-28
No generic	-0.28	3.358e-27
No glycans no generic	-0.28	9.476e-25
Reaction Graph		
KEGG kgml	0.14	3.272e-08
No glycans	0.18	2.923e-11
No generic	0.13	3.032e-07
No glycans no generic	0.17	1.106e-10

Table 4: correlation coefficients between node degree and cluster coefficients

3.1.3 RPAIR data

RPAIR data [18] stored in KEGG ligand was used for metabolic network construction and again compared to the original KEGG kgml dataset 3.1.1. We can thereby observe the features of a network where currency metabolites are supposedly accounted for.

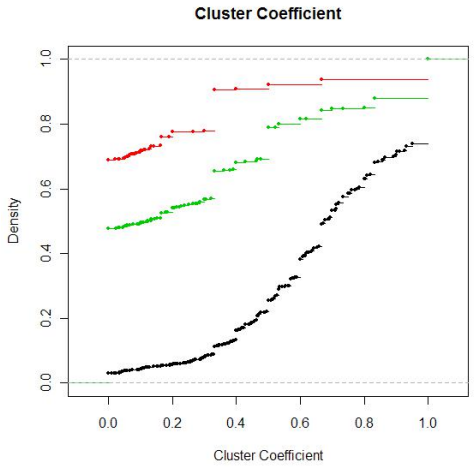
The **RPAIR** dataset was created by again collecting all reactions from the "eco01100.kgml" file. This time, to retrieve the reaction equation, we checked the reaction file in KEGG LIGAND and excluded all reactants which were not part of one main reactant pair. Reactions were split if the main reactant pairs of one reaction do not add up to a full reaction. This can be illustrated by the following example: if, in a reaction with substrates C1 and C3 and products C2 and C4, there are only two main reactant pairs: C1 - C2 and C3 - C4 and there is no main pair connecting C1 or C2 with C3 or C4 the reaction was split into two reactions ($C1 \leftrightarrow C2$ and $C3 \leftrightarrow C4$). Generic molecules were excluded from the RPAIR set and glycans do not appear in it.

The **RPAIR2** dataset was created almost exactly the same way as the RPAIR. The only difference is that reactions are not split as explained above. So if, for one reaction, there are only two main pairs C1 - C2 and C3 - C4 the resulting reaction will be $C1 + C3 \leftrightarrow C2 + C4$.

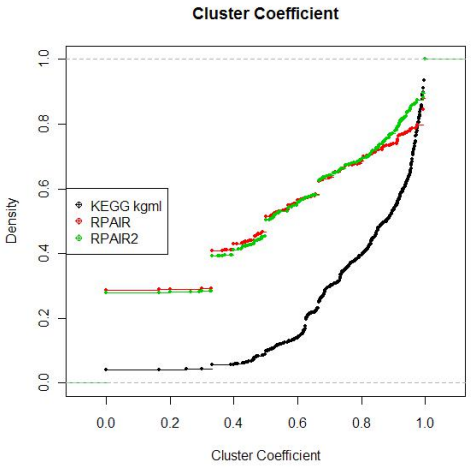
	KEGG kgml	RPAIR	RPAIR2
Size (nodes)	3067	2831	2716
Reaction nodes	1477	1592	1477
Compound nodes	1590	1239	1239
Edges	6177	3096	3096

Table 5: Number of the nodes and edges in the different networks.

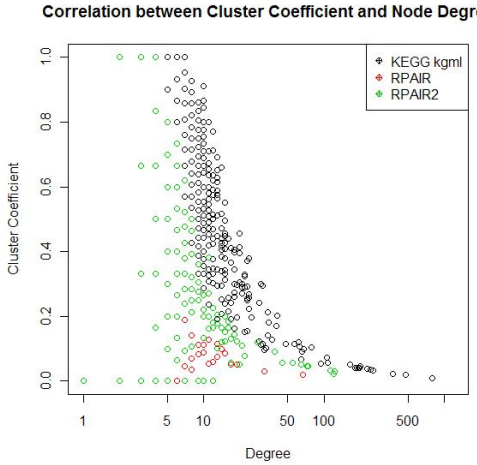
Compared to all other networks analyzed so far, the difference between the numbers of reaction and compound nodes is highest in the two RPAIR networks (see table 5).



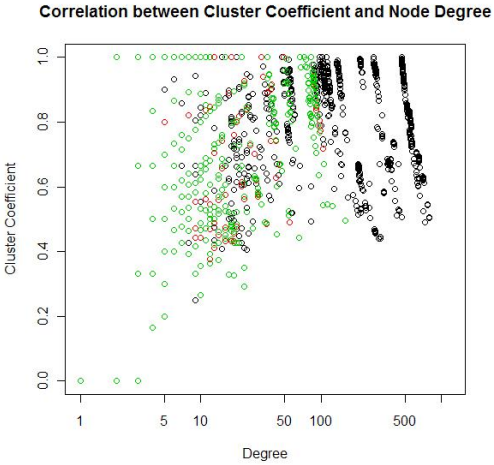
(a) cluster coefficient in the compound graph



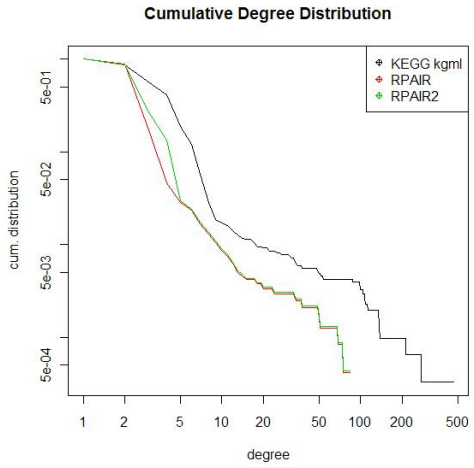
(b) cluster coefficient in the reaction graph



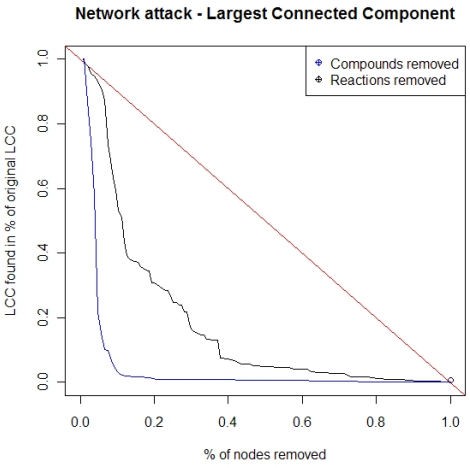
(c) cluster coefficient plotted against node degree in the compound graph



(d) cluster coefficient plotted against node degree in the reaction graph



(e) cumulative degree distribution



(f) network attack RPAIR

Figure 9: Different measurements of the curated data sets

	Correlation coefficient	p-value
Compound Graph		
KEGG kgml	-0.284	7.886e-31
RPAIR	0.061	0.048
RPAIR2	0.005	0.86
Reaction Graph		
KEGG kgml	0.14	3.272e-08
RPAIR	0.60	2.2e-16
RPAIR2	0.57	2.2e-16

Table 6: correlation coefficients between node degree and cluster coefficients

351 compounds are not found any more in the network (1590 - 1239) because they are never part of a "main" reactant pair. Furthermore some reactions were split in the RPAIR network, which increased the number of reactions. The ratio of nodes to edges also changed from approximately 1:2 to almost 1:1.

The structural measures of the RPAIR networks are also quite different from all other networks analyzed so far (here they are only compared to the raw KEGG kgml data):

The cumulative degree distribution (figure 9(e)) shows that degrees of nodes are generally lower in the RPAIR graphs. This is expected because we were trying to eliminate the problem of currency metabolites, which have a lot of connections. Also, The logarithmic plotting of this distribution seems to be straighter than for the other networks. This might be interesting for the discussion about power law distribution [20] and scale-freeness of the network although it is not the question here.

Looking at the cluster coefficients in the compound graph (figure 9(a)) of both RPAIR datasets we obtain lower clustering than in the other datasets with a lot of nodes at the value of about 0. This makes somewhat more sense with respect to the following: Nodes contribute to the cluster coefficient when three of them (A, B, C) are interconnected with each other. Chemically speaking, this situation corresponds to compound A being converted into compound C by one reaction (one arc in the graph) but also in two reactions ($A \rightarrow B$ and $B \rightarrow C$). If the reactions are elementary, meaning that they cannot be broken down into subreactions, this should not be the case. The reason why we still find connected triangles is the fact that not all reactions are elementary reactions. In paths (sequences of nodes) nodes with a cluster coefficient of 0 are expected. They have two neighbors which are not connected to each other. We can further see that RPAIR one data is less clustered than RPAIR2 which is intuitively explainable. Since we have less nodes but the same amount of edges in RPAIR2 (table 5), we can expect more interconnections.

Compared to what is observed in the compound graph, the RPAIR reaction graphs show relatively high clustering (figure 9(b)). So if we use RPAIR data, where currency metabolites are expected to pose less of a problem, we can observe differences between the compound and reaction graph with respect to clustering.

The interpretation of the correlation between cluster coefficients and degrees is somewhat problematic looking at figures 9(c) and 9(d). We can however find a correlation for the RPAIR reaction graphs (table 6) but no more in the compound graph. This is the inverse situation as it was the case for all other networks seen so far.

Looking at the measurement of network vulnerability we can see that the RPAIR network still shows the same vulnerability against removal of compounds but it fragments faster by removal of reaction nodes (figure 9(f)) compared to raw KEGG kgml (figure 6(f)). However reaction removal is still less critical to network size than compound removal. Note that, as explained above, if a reaction is removed here, it only concerns one of multiple reactions corresponding to one KEGG reaction entry because they were split earlier. To avoid this we continued our analysis of network attack with the RPAIR2 network.

3.2 RPAIR 2 Network attack and error tolerance

Motivated by our finding that metabolic networks are more tolerant to reaction node removal than to compound node removal (figure 9(f)) we analyzed this feature of the network in more depth.

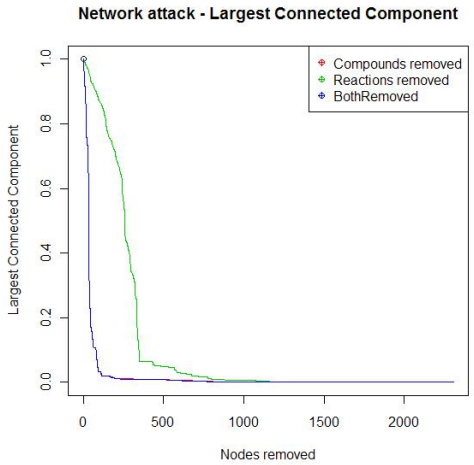
We attacked the RPAIR2 network (from section 3.1.3) by deleting nodes in order of their degree (figure 10(a)) and betweenness (figure 10(b)). In addition to this network attack tolerance we also analyzed network error tolerance by random removal of nodes (figure 10(c)). Albeit random removal leads to different curves every time it is repeated (unlike centrality directed removal), we here only show one curve for each possibility of nodes to remove (compounds, reactions or both).

We then wanted to understand if the observed tolerance or vulnerability of the network is caused by the structural arrangement of the nodes. This was done by comparing the network to randomized networks. Creating appropriate randomized bipartite graphs is not a task with an obvious solution. We here used network rewiring (section 6.4) in a similar way it was done elsewhere [25] in the context of flux balance analysis. Note that every time we do rewiring the resulting network will be a different one. So we deal with a population of different possible arrangements which we compare to the arrangement of the biological network.

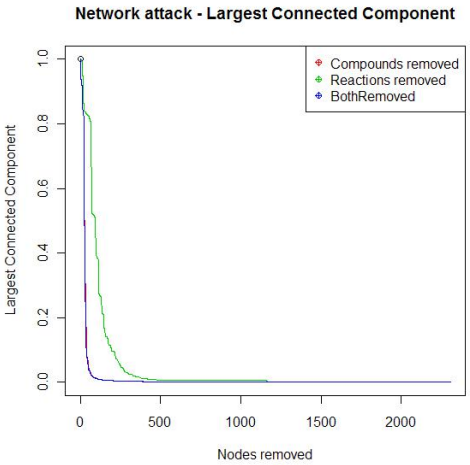
First we analyzed the outcome of our rewiring algorithm. We compared the degree distribution of the original, biological, network (figure 11(a)) to the degree distributions in some rewired networks (figures 11(b) and 11(c)). Since they are almost the same we presume that we successfully preserved the degree distribution of the network. However, we found that some edges were lost by this procedure because sometimes we would introduce two edges between the same endpoints which is not allowed in the graph. The probability of this event is relatively small, so the loss of edges was never bigger than one percent of the size of the original network. To understand the created differences in structure we then also compared the betweenness distribution (figures 12(a) and 12(b)). Judging from these plots we, apart from preserving the degree distribution of the original network, also created networks with a similar overall structure.

We then continued by actually comparing the curves obtained by network attack of biological and random networks [25] (table 7). For both cases of attack (degree, figure 13, and betweenness, figure 14), subfigures a and c show the curves of the biological network compared to some rewired ones. To be able to measure the difference between the curves of rewired networks and actual networks we also plotted the difference of the mean of the rewired ones (subfigures b and d). To account for the variation between the random networks we then further performed a statistical test (section 6.3). This test measures if the difference between the area under the curves of the random networks and the curve under the biological network is significant. The resulting p-values is shown in the subfigures a and c.

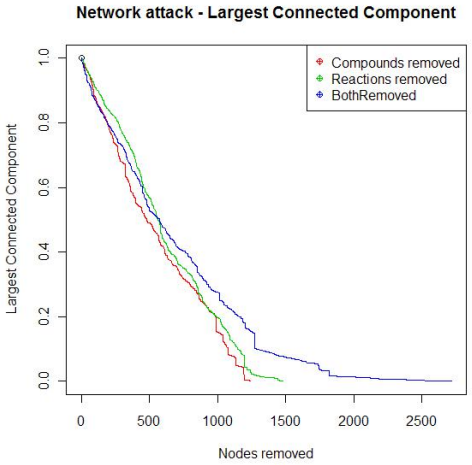
There are some conclusions we can draw from these plots:



(a) Nodes removed in order of degree: Note: The curves for all nodes and compound nodes are overlapping.

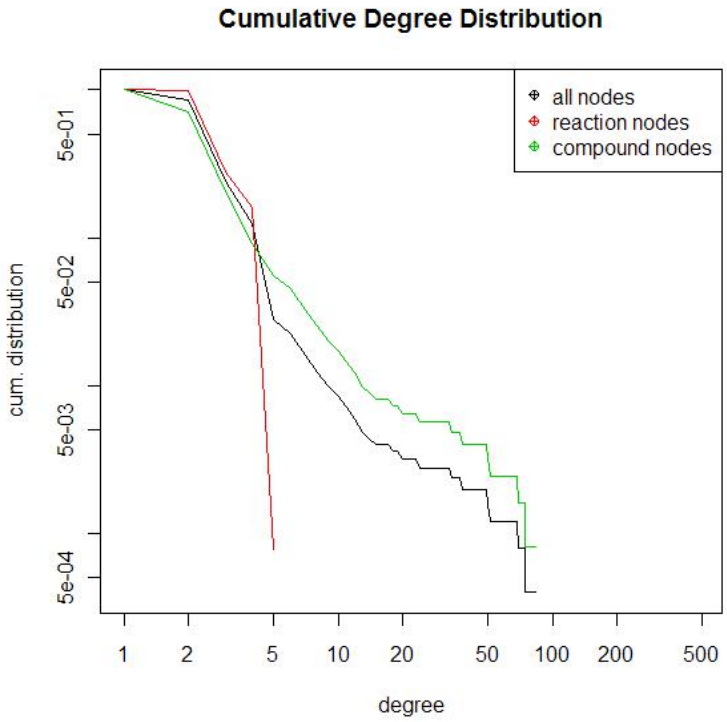


(b) Nodes removed in order of betweenness: Again, the curves for all nodes and compound nodes are overlapping.

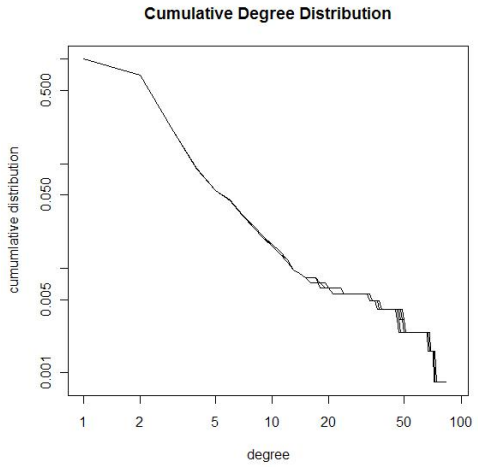


(c) Random nodes removal.

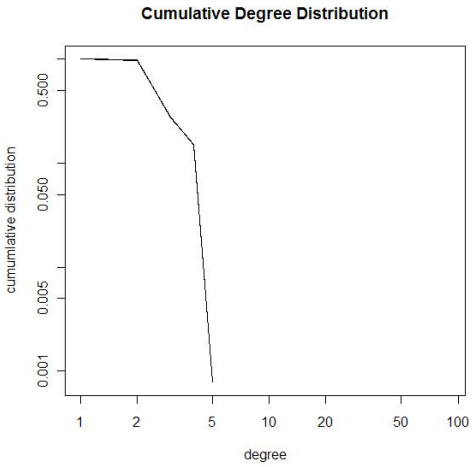
Figure 10: Network attack and error tolerance of the RPAIR network



(a) biological network

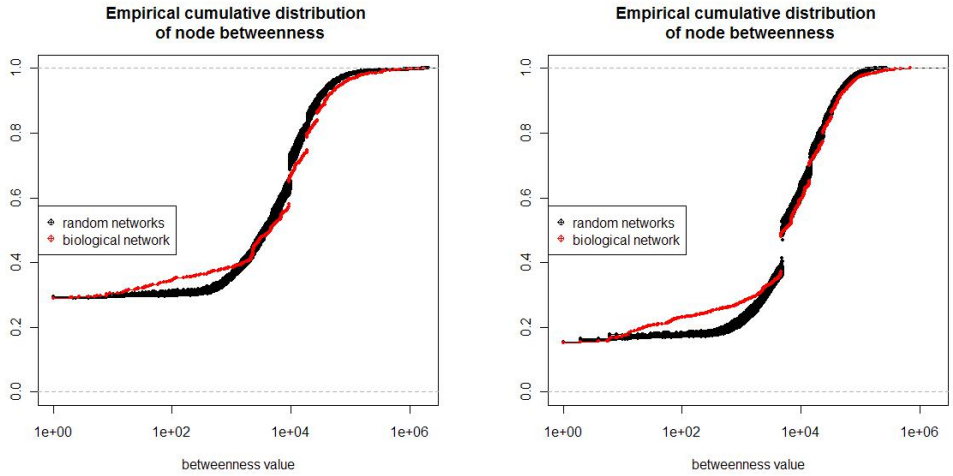


(b) Random network degree distribution of compounds



(c) Random network degree distribution of reactions

Figure 11: Degree distribution of the real network and the rewired ones



(a) Betweenness distribution of compounds (b) Betweenness distribution of reactions

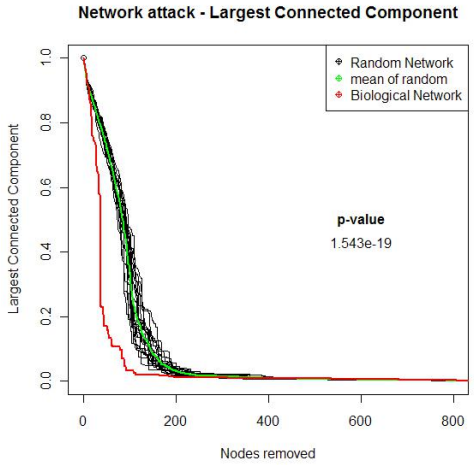
Figure 12: Betweenness distribution of the real network and the rewired ones

Attacking	both	compounds	reactions
betweenness	10	10	7
degree	20	20	20

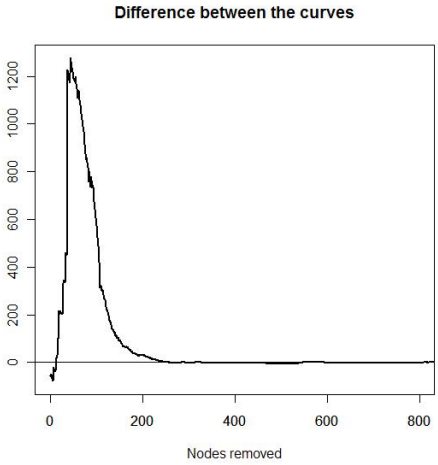
Table 7: table summarizing the number of rewired networks used in each analysis

First we can see that the networks are more tolerable to random rather than directed removal of nodes (figure 10). This goes along with the observations of [1] who concluded the same after comparing exponential to power-law networks. Judging from figure 11(a) our metabolic network's degree distribution does not quite fit a power-law distribution (as it was shown before for metabolic networks [20]). However, we can observe that a perfect fit is not necessary to yield a network which is tolerant to network error and vulnerable to network attack.

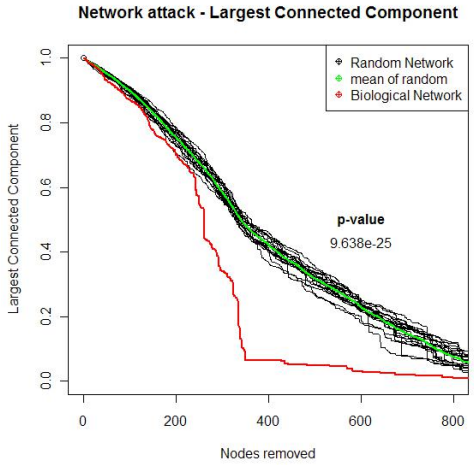
Since our network is bipartite we can also focus on roles of the two different species, reactions and compounds, in the network. In the case of network error (figure 10(c)) we do not observe any difference. However when attacking the network, we see that the removal of reactions has a less destructive effect on the network than removing compounds, which destroys the network almost instantly . Both, degree (figure 10(a)) and betweenness (figure 10(b)) as centrality measurement, lead to this conclusion. Obviously compounds have higher degrees than reaction nodes, meaning that one molecule can take part in many reactions while reactions have far less substrates and products. In the graph this means that reactions are connected to a maximum of 6 compounds, while some compounds are connected to many more reaction nodes (see figure 11(a)). For betweenness measurements the observation that compounds are more important than



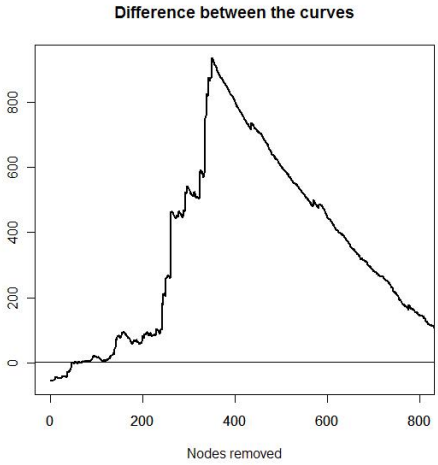
(a) Only compound nodes were removed. Random networks (black) compared to biological ones (red).



(b) Only compound nodes were removed. The difference between the mean of the random networks minus the biological one.

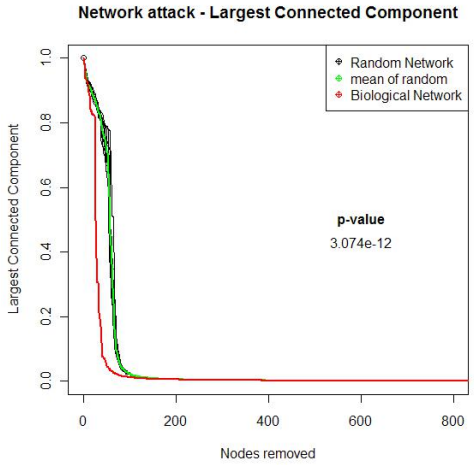


(c) Only reaction nodes were removed. Random networks (black) compared to biological ones (red).

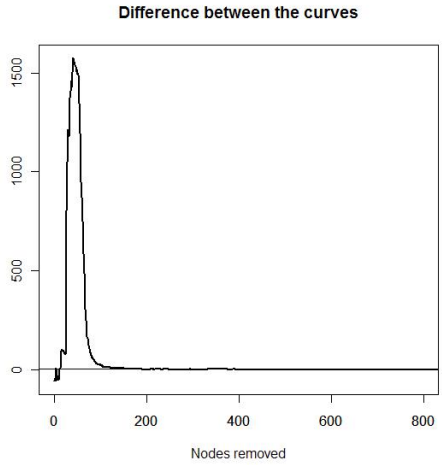


(d) Only reaction nodes were removed. The difference between the mean of the random networks minus the biological one.

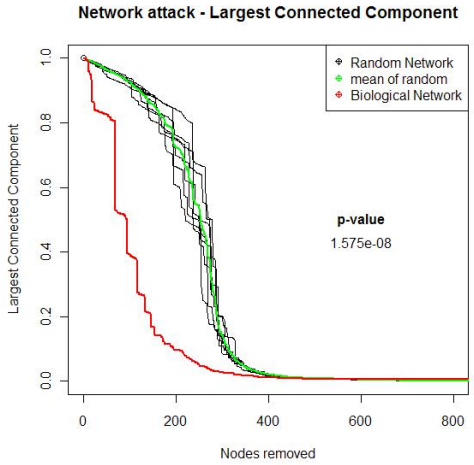
Figure 13: Random Networks of RPAIR2: Nodes were removed in order of their degree.



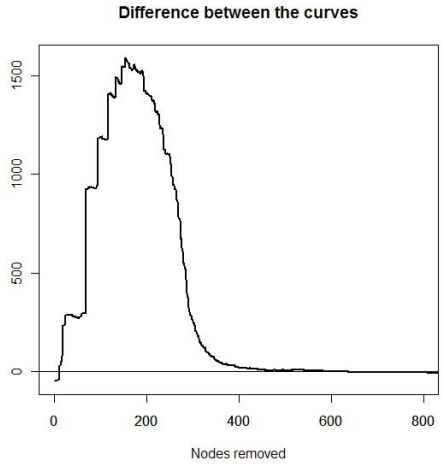
(a) Only compound nodes were removed. Random networks (black) compared to biological ones (red).



(b) Only compound nodes were removed. The difference between the mean of the random networks minus the biological one.



(c) Only reaction nodes were removed. Random networks (black) compared to biological ones (red).



(d) Only reaction nodes were removed. The difference between the mean of the random networks minus the biological one.

Figure 14: Random Networks of RPAIR2: Nodes were removed in order of their betweenness.

reactions is less obvious (see discussion, section 4.2).

We can further confirm the conclusions about the importance of compound nodes by comparing the curves of the restricted removal of compounds nodes to the unconstrained removal of both types of nodes in figure 10(a) and 10(b). The curves are actually overlapping so we see that if there is no restriction on what nodes to remove, mainly compound nodes will be removed and therefore the curves are very similar. In degree-directed attack it is again obvious that compound nodes will be selected first, because their degree value is much higher. The same is not true for betweenness judging from the initial betweenness values (figures 12(a) and 12(b)). However these can change a lot when they are recalculated after nodes are removed.

If we compare the figures with degree versus betweenness to measure node centrality we can see that for each pair of curves (figures 13(a) and 14(a) as well as 13(c) and 14(c)) bigger parts of the network are broken off by betweenness directed rather than degree directed node removal. However this can only really be observed for the removal of reactions because when deleting compounds with either measure (degree or betweenness) the network breaks down almost instantly as explained above.

By comparing biological to random networks we make the rather unexpected discovery that the biological network is less tolerant to node removal. Similar to the biological network, deletion of compounds (figures 13(a) and 14(a)) breaks down random networks very quickly. This is the case for degree or betweenness as a centrality measurement. Still they show a higher resistance than the biological network. Only looking on the figures on the left we could think that this difference is marginal, however the corresponding plots of the difference between the curves (figures on the right) show that the biological network actually breaks down a lot faster. After the removal of about 100 nodes the difference between the networks is more then 1000 in size (the whole network has 2716 nodes). As shown by the p-value we can see that the difference in area under the curves are clearly significant. When reaction nodes are removed in order of their degree (figure 13(c)), we cannot make a similar statement: Even though the statistical measure shows that the difference in area under the curve is significant more than 200 nodes have to be removed for this effect to become visible. On the contrary, looking at the curve for betweenness (figure 14(c)), we see a greater difference. The random networks seem to be a lot more resistant to node removal in this case. The statistical test is slightly weaker but still significant. The weak statistic is mainly due to the small amount of random networks here (see table 7).

4 Discussion

4.1 Comparing datasets

The comparison of the four datasets (KEGG from the kgml file and the three datasets from different databases derived from only EC annotation, see section 3.1.1) do not show interpretable differences. All the measurements taken are quite similar. Recall the difference between KEGG kgml and KEGG EC is that KEGG kgml contains glycans and uncatalyzed reactions while KEGG EC does not. Both contain generic molecules that appear in their reactions. MetaCyc and RhEA have some glycans and generic molecules but no uncatalyzed reactions. The reason why MetaCyc and RhEA might be smaller is because of the differences between these databases regarding the link between EC annotation and reactions (see section 2.1). While in KEGG EC all possible reactions linked to an EC number are taken, MetaCyc and RhEA seem to be more restrictive. This is one problem using the approach of mapping EC numbers to reactions. There can be multiple EC numbers for one reaction and multiple reactions for one EC number. Furthermore the EC number is sometimes not sufficient to specify exactly which reactions can or cannot be catalyzed by the enzyme.

Comparing KEGG kgml to different curation methods (section 3.1.2) used by some authors [12, 21] we see no effect on the network's structure and only a small effect its size. Maybe glycans and generic molecules are just outside of the core of the network and therefore have little influence on its structure.

The networks constructed from RPAIR data (section 3.1.3) show highly different features. A lot of edges and some compounds are removed from the graph. This results in lower clustering which is expected if we assume that the network is made up out of linear sequences of compounds and reactions. RPAIR data is probably best suited for network construction because the problem of currency metabolites is directly tackled. For example, it has been shown that path finding in an RPAIR graph increases the chance of identifying already known paths which was less likely with other methods [7].

4.2 Network vulnerability

To our knowledge, the difference in network attack tolerance (first introduced by [1]) to the deletion of compound or reaction nodes was not yet reported. While the network is fragmented quickly by the deletion of compound nodes the same happens more slowly when deleting reaction nodes. This is intuitive but not obvious. Compounds have far more connections than reactions but reactions are still needed to connect the compounds. The difference was first observed on the raw KEGG kgml network (figure 6(f)) when nodes were deleted in order of their degree. The same was then also observed when deleting some currency metabolites (figure 7(a)) or in the RPAIR network (figure 9(f)).

So does this observed difference have a biological meaning? What does it mean to delete compound or reaction "nodes"? Compound nodes are hard to actually remove

from a cell. They can only be modified to other molecules. In contrast to compounds, reactions can be removed from the cell. If the reaction is catalyzed by an enzyme and would be very slow otherwise, we can consider it to be removed if we remove the catalyzing protein [19, 20].

To better understand the different roles of compound and reaction nodes we then continued the analysis by also deleting nodes in order of their betweenness and by random selection. We concentrated only on the RPAIR2 network described in section 3.1.3 because there, reactions are not split. So if one reaction is removed, it resembles more or less a whole protein of gene being removed (see section 2.3 for the limitations of this conclusion). From these studies, we still observe that removing compounds is more efficient to break down the network. Furthermore we observe that directing the attack against high betweenness nodes disconnects the network very fast. Random removal, on the contrary, is a very weak attack strategy. So finally, we conclude that the best way to break down a metabolic network is to remove compounds with a high betweenness. For our network this means that the compounds are the important connectors between the subnetworks. This result is not quite intuitive if we look at the distribution of betweenness of compounds (figure 12(a)) and reactions (figure 12(b)). There is very little difference between the two but it is still causing more damage to remove compounds than reactions with high betweenness values.

On the contrary, the finding that a betweenness directed attack is breaking down the network faster than a degree directed one is rather obvious. Node degree is a rather local measure concerning only one node and its neighbors while betweenness is truly global in the sense that it takes into account all shortest paths in the whole network. From this we can draw conclusions about modularity. If we assume the biological network is built up of modules with a lot of edges inside the module but few between the modules then we would think that nodes which connect these modules have a high betweenness.

After these finding we continued by comparing random and real networks in respect to their vulnerability to attack. This way we can see the effect of structural rearrangements in the network. Originally we assume that, through evolution and selection, the network is organized in a way which limits the effect of network attack. It is astonishing that our metabolic network does not show this property. It is actually clearly suboptimal with regard to tolerance to node attack.

But what does it mean to compare metabolic networks of a cell to random ones? Chemically, these (randomized) arrangements of reactions and compounds are not feasible at all. Only certain molecules can be converted into other ones. In contrast to other networks (cell signaling or protein - protein interaction) a metabolic network has very strict chemical constraints and the given structure cannot be changed. The reactions also have to be in a given order and follow energetic rules. So, against the theory that real networks are comparatively resistant, we could hypothesize that others, for example minimization of energy consumption and other chemical constraints, are more important driving forces in evolution than to create a resistant network.

5 Conclusions

Clearly, it is not easy to link the analysis of metabolism with graph theory to the validation of the results in laboratory experiments. Therefore different approaches have to be found to be able to validate and compare results. One attempt was to show how well different algorithms perform to reproduce existing biological knowledge. For example, a clustering algorithm was used to detect major biochemical pathways [9] and a path-finding algorithm was compared to known valid paths in the metabolic network [5].

In the present article we took a different approach by checking the effect of different data representation and curation effects on the created network. Surprisingly we did not find strong changes of the network structure when varying databases used or curation efforts for glycans or generic molecules. On the contrary, tackling currency metabolites with the use of RPAIR data proofed to strongly effect the network. So it is obvious that the issue of currency metabolites has to be addressed when dealing with metabolic networks.

In the second part of our study we observed a high importance of compounds of high betweenness in the network. From this information we assume that they are the connectors between modules in the network which makes sense with respect to the fact that to remove compounds from the network is far harder if not impossible compared to remove reactions. We think that this very structural information can be interesting mainly for the detection of modules in the network.

Finally, we think that our network representation has some issues if we want to retrieve non-structural information like identifying biochemical paths or making assumptions about design principles of complex networks: The main argument for this thought is that, when comparing real and random networks, we found that the real network is actually significantly more vulnerable than the random ones. This being contrary to our naive expectation, we hypothise that the shape of the network is determined by other factors rather than tolerance to topological attack and errors. Similarly it was also proposed elsewhere ([24] and references therein) that other information, such as reaction speed, energetic cost and the possibility of regulating steps, should be integrated in a model to study the metabolic network and the biochemical paths found therein. In addition we also stress the fact that the structure of the network is strongly constrained by chemical rules and cannot be freely reorganized which will also influence the structure of the network.

6 Methods

6.1 Construction of the metabolic network

An array of PERL scripts was used for the different data-sets retrieving all the reactions which take place in E. Coli, their substrates, products and if it is considered as being reversible (although finally all reactions were said to be reversible). For KEGG data the names for the compounds could also be retrieved. All of the information was then stored in a xml file.

A Java program was then used to create an unweighted, bipartite and undirected network representing this information. The xml file was read with the xstream package (xstream.codehaus.org) to create reaction and compound objects. From these, a network was created with the help of the jung2 package (jung.sourceforge.net).

6.2 Network analysis

Because most networks were not completely connected after construction the biggest one was selected for further analysis. However, this has very little effect on the data since the second biggest network was always very small. The jung2 package in Java was then used for different graph manipulations and to measure the different parameters. These measurements were exported to R where statistical analysis and plotting was done.

6.3 Measurements

In order to calculate the **cluster coefficient** one has to decide to analyze either the compound or reaction network, as already explained in the introduction (section: 2.5). Once the bipartite graph is transformed into a compound or reaction graph the value of the cluster coefficient for each node can be easily computed by equation 13. For one node, the cluster coefficient (C_i) is the division of the number of all existing links between its neighbors (E_i) by the number of all possible links between them ($k_i(k_i - 1)$; k_i is the number of neighbors k of node i). The empirical cumulative degree distribution (ecdf() in R) of the cluster coefficients was then plotted for visual comparison. For every value it tells us the fraction of nodes with a cluster coefficient smaller or equal than this value.

$$C_i = \frac{2E_i}{k_i(k_i - 1)} \tag{13}$$

When plotting the **degrees of nodes against their cluster coefficients** one faces the same fact as above. Accordingly one can only plot node degrees against their cluster coefficients in a reaction or compound graph.

To plot the tolerance of the network against attack and error the size of the **largest connected component** was used in the form $\frac{biggest\,existing\,subnetwork}{biggest\,original\,subnetwork}$. It was either plotted against the number of nodes removed from the network or against the fraction $\frac{removed\,nodes}{all\,nodes}$.

Centrality measures like betweenness and degree were recalculated after each node removal. Using the betweenness values of the initial network (without recalculating it every time) was tried but it was far weaker in breaking down the network as it was also shown elsewhere [11].

For the **cumulative degree distribution** the degree values for all nodes that is the number of neighbors were read. The value of the cumulative degree distribution of degree k is then the fraction of nodes that have a degree equal or higher than k [20].

The **empirical cumulative distribution of betweenness** tells us for each betweenness value how many percent of the nodes have a value smaller than that one. The function "ecdf()" in R is used for that purpose.

Statistical test to compare the **area under the curves**: The area under the curve is normalized to be a value between 0 and 0.5. The p-value comes from a one sample t-test (also implemented in R) with the null hypothesis that the true mean of the population of areas under the attack curves of the random networks is equal to the area under the curve of the biological network.

6.4 Network rewiring

Contrary to [25], our simplified rewiring algorithm was implemented as follows: After selecting two edges at random, their endpoints (one compound and one reaction) are swapped. If, before rewiring, edge E1 connects compound C1 to reaction R1 and E2 connecting C2 to R2, we afterwards find C1 connected to R2 and C2 connected to R1 with two new edges. This swapping was done so each edge is changed once. The outcome of the rewiring algorithm is explained in section 3.2.

7 Acknowledgments

I want to mainly thank my supervisor, Frederique Lisacek, for the guidance of my work and her enriching ideas and questions, Christophe Charpilloz, who was always ready for very practical advices of all sorts and Anne Morgat for her ideas and tips on how to use biological data and databases. I would further like to thank Kristian Axelson for getting me started in investigating metabolism, Jean-Luc Falcone for some very clarifying discussions and Frederic Nikitin for his help on how to smoothly solve program development issues. Of course many of my thanks also go to all the teachers and assistants of the Master in Proteomics and Bioinformatics, especially its coordinator Patricia Palagi, for giving me the opportunity to discover the use of computation in biology. Finally I want to thank my friends and family for their constant support and feedback.

References

[1] R. Albert, H. Jeong, and A. Barabasi, *Error and attack tolerance of complex networks*, Nature, 406 (2000), pp. 378–382.

[2] M. Arita, *The metabolic world of escherichia coli is not small*, Proceedings of the National Academy of Sciences of the United States of America, 101 (2004), pp. 1543–1547. PMID: 14757824.

[3] R. Caspi, H. Foerster, C. A. Fulcher, P. Kaipa, M. Krummenacker, M. Latendresse, S. Paley, S. Y. Rhee, A. G. Shearer, C. Tissier, T. C. Walk, P. Zhang, and P. D. Karp, *The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of Pathway/Genome databases*, Nucleic Acids Research, 36 (2008), pp. D623–D631. PMID: 17965431 PMCID: 2238876.

[4] C. Chaouiya, *Petri net modelling of biological networks*, Briefings in Bioinformatics, 8 (2007), pp. 210–219. PMID: 17626066.

[5] D. Croes, F. Couche, S. J. Wodak, and J. van Helden, *Inferring meaningful pathways in weighted metabolic networks*, Journal of Molecular Biology, 356 (2006), pp. 222–236. PMID: 16337962.

[6] J. S. Edwards, R. U. Ibarra, and B. O. Palsson, *In silico predictions of escherichia coli metabolic capabilities are consistent with experimental data*, Nat Biotech, 19 (2001), pp. 125–130.

[7] K. Faust, D. Croes, and J. van Helden, *Metabolic pathfinding using RPAIR annotation*, Journal of Molecular Biology, 388 (2009), pp. 390–414.

[8] P. Gerlee, L. Lizana, and K. Sneppen, *Pathway identification by network pruning in the metabolic network of escherichia coli*, Bioinformatics, 25 (2009), pp. 3282–3288.

[9] R. Guimera and L. A. N. Amaral, *Functional cartography of complex metabolic networks*, Nature, 433 (2005), pp. 895–900.

[10] T. Handorf, N. Christian, O. Ebenhh, and D. Kahn, *An environmental perspective on metabolism*, Journal of Theoretical Biology, 252 (2008), pp. 530–537. PMID: 18086477.

[11] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han, *Attack vulnerability of complex networks*, Physical Review. E, Statistical, Nonlinear, and Soft Matter Physics, 65 (2002), p. 056109. PMID: 12059649.

[12] M. Huss and P. Holme, *Currency and commodity metabolites: their identification and relation to the modularity of metabolic networks*, IET Systems Biology, 1 (2007), pp. 280–285. PMID: 17907676.

[13] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A. L. Barabsi, *The large-scale organization of metabolic networks*, Nature, 407 (2000), pp. 651–654. PMID: 11034217.

[14] B. H. Junker and F. Schreiber, *Analysis of biological networks*, Wiley-Interscience, 2008.

[15] M. Kanehisa and S. Goto, *KEGG: kyoto encyclopedia of genes and genomes*, Nucleic Acids Research, 28 (2000), pp. 27–30. PMID: 10592173.

[16] M. Kanehisa, S. Goto, M. Furumichi, M. Tanabe, and M. Hirakawa, *KEGG for representation and analysis of molecular networks involving diseases and drugs*, Nucleic Acids Research, 38 (2010), pp. D355–360. PMID: 19880382.

[17] M. Kanehisa, S. Goto, M. Hattori, K. F. Aoki-Kinoshita, M. Itoh, S. Kawashima, T. Katayama, M. Araki, and M. Hirakawa, *From genomics to chemical genomics: new developments in KEGG*, Nucleic Acids Research, 34 (2006), pp. D354–357. PMID: 16381885.

[18] M. Kotera, R. Yamamoto, K. Tonomura, M. Hattori, T. Komeno, S. Goto, J. Yabuzaki, and M. Kanehisa, *RPAIR: a reactant-pair database representing chemical changes*.

[19] N. Lemke, F. Heredia, C. K. Barcellos, A. N. dos Reis, and J. C. M. Mombach, *Essentiality and damage in metabolic networks*, Bioinformatics, 20 (2004), pp. 115–119.

[20] G. Lima-Mendez and J. van Helden, *The powerful law of the power law and other myths in network biology*, Molecular bioSystems, 5 (2009), pp. 1482–1493. PMID: 20023717.

[21] H. Ma and A. Zeng, *Reconstruction of metabolic networks from genome data and analysis of their global structure for various organisms*, Bioinformatics, 19 (2003), pp. 270–277.

[22] O. Mason and M. Verwoerd, *Graph theory and networks in biology*, q-bio/0604006, (2006).

[23] M. E. J. Newman, *Assortative mixing in networks*, Physical Review Letters, 89 (2002), p. 208701. PMID: 12443515.

[24] F. J. Planes and J. E. Beasley, *A critical examination of stoichiometric and path-finding approaches to metabolic pathways*, Brief Bioinform, 9 (2008), pp. 422–436.

[25] A. G. Smart, L. A. N. Amaral, and J. M. Ottino, *Cascading failure and robustness in metabolic networks*, Proceedings of the National Academy of Sciences, 105 (2008), pp. 13223–13228.

[26] J. Zhao, H. Yu, J. Luo, Z. Cao, and Y. Li, *Complex networks theory for analyzing metabolic networks*, Chinese Science Bulletin, 51 (2006), pp. 1529–1537.