Hierarchical Organization of Modularity in Metabolic Networks

Supporting Online Material

Contents

1. Network Models

1.1 Scale-free and Modular Networks

1.1.1 Models

The scale-free model (Fig. 1a, article)

The scale-free model is generated from a small initial core of connected nodes with the addition of a new node in each timestep[1]. This new incoming node connects m undirected and unweighted links to the existing nodes following preferential attachment. The probability Π_i of the new node to attach to node i is proportional to the number of links node i already has:

$$\Pi_i = \frac{k_i}{\sum k_i}.$$

The modular model (Fig. 1b, article)

The modular model is a generalization of the Erdős-Rényi random graph[2]. Instead of starting with N nodes and connecting all nodes with probability p, we start with m groups of nodes of relative sizes f_1, f_2, …, f_m ($\sum f_i = 1$). We connect all nodes within a group (module) with probability q and connect all pairs belonging to different groups with probability $p << q$. This model gives rise to a random, but inherently modular network. A closely related model was proposed recently in Ref [3].

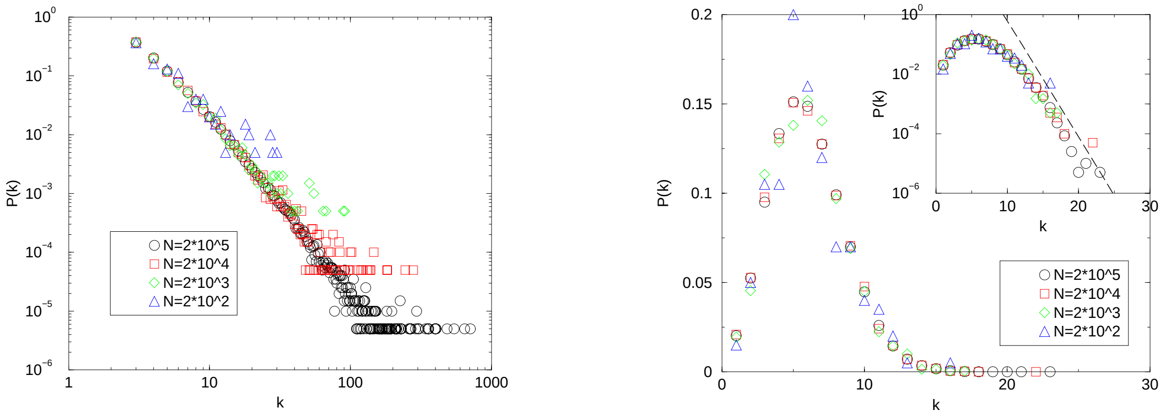
1.1.2 Properties of the model networks

Degree distribution

The degree distribution is defined as the probability $P(k)$ that a randomly chosen node in the network has a degree (number of links) k.

- For the scale-free model $P(k) \sim k^{-3}$ (Fig. S1a)[1]. Such power-law dependence means that there is no characteristic scale for the degree values; while an average connectivity can be defined, the second momentum of the distribution diverges.

- For a random network the degrees follow a Poisson distribution, characterized by an exponential tail. The average connectivity for such a graph has well-defined average value, and the second moment of the distribution is finite. The degree distribution of the modular network model (Fig. 1b, article) has an exponential tail, the modularity having little affect on degree distribution. Thus its degree distribution is similar to that of a random graph (Fig. S1b).



(a) Scale-free networks.

(b) Modular networks.

Figure S1. Degree distributions of model networks with different sizes ($< k >= 6$).

Clustering coefficient

The clustering coefficient of a node gives the probability that its neighbors are connected to each other. It is defined as[4]

$$C_i = \frac{2n}{k_i\,(k_i - 1)},$$

where n denotes the number of direct links connecting the k_i nearest neighbors of node i. The clustering coefficient is close to one for a node at the center of a highly interlinked cluster, while it is zero for a node that is part of an only loosely connected group. Therefore C_i, averaged over all nodes i, is a measure of the network's potential modularity, since it offers a measure of the degree of interconnectivity in the neighborhood of each node. For example, a node whose neighbors are all connected to each other has $C = 1$ (Fig. S2, left), while one with no links between its neighbors has $C = 0$ (Fig. S2, right). A network for which the average clustering coefficient is large is expected to be highly interconnected, a sign of clustering.

The average clustering coefficient of the scale-free model generated for different network sizes is known to decrease as $N^{-3/4}$ (See Fig. 2b from article).

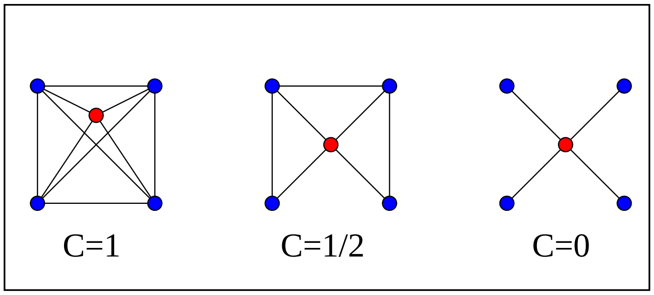


Figure S2. Definition of the clustering coefficient. The numbers show the clustering coefficient of the central node.

For a random network all nodes and links are statistically identical, thus the average clustering coefficient of a node is $< C > = p = < k > /N$, since the probability that two of a node's neighbors are linked is the same as the probability that any pairs of nodes are linked. The modular model has random network characteristics at two superimposed levels: a node is part of its internally homogeneous module, as well of a more dilute random graph representing the whole network. For the j-th module representing a fraction f_j of the whole network

$$< k >_j = [qf_j + p(1 - f_j)]\, N.$$

Each node's neighbors in this module j have link contributions of roughly $q\, f_j^2\, N^2$ from its neighbors from inside the module and of $p\,(1{-}f_j)^2\, N^2$ from its neighbors from outside. Thus

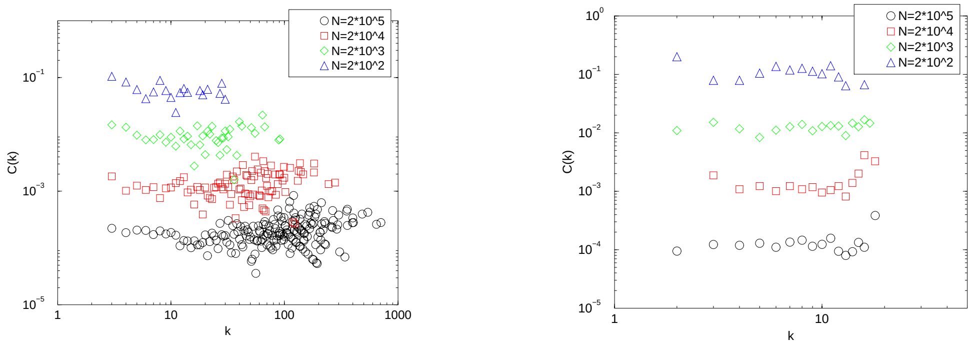
$$< C >_j = \frac{qf_j^2 + p(1 - f_j)^2}{[q\, f_j + p\,(1 - f_j)]^2}.$$

Thus, for a constant $< k >$ ($p\, N =$ const.; $q\, N =$ const.)

$$< C > = \sum f_j \;< C >_j \sim N^{-1}$$

similar to the behavior observed for the Erdős-Rényi network.

As the value of $< k >$ does not reflect the crucial difference between the two network models, the average clustering coefficient does not reflect the inherent structural differences between the two networks. We define the function $C(k)$ as the average clustering coefficient of nodes with degree k. Measurements of this function show that $C(k)$ is independent of k for both the scale-free and the modular networks, indicating that small nodes and large hubs display similar clustering properties (Fig. S3).



(a) $C(k)$ for scale-free networks.

(b) $C(k)$ for modular networks.

Figure S3. For scale-free and modular networks $C(k)$ is independent of k, in contrast with the $C(k) \sim k^{-1}$ observed for the metabolic networks (Fig. 2c-f, article) and the hierarchical model (Fig. S5b).

1.2 Hierarchical scale-free network

1.2.1 Construction

In order to combine a modular structure with a scale-free topology we propose a simple deterministic hierarchical model.

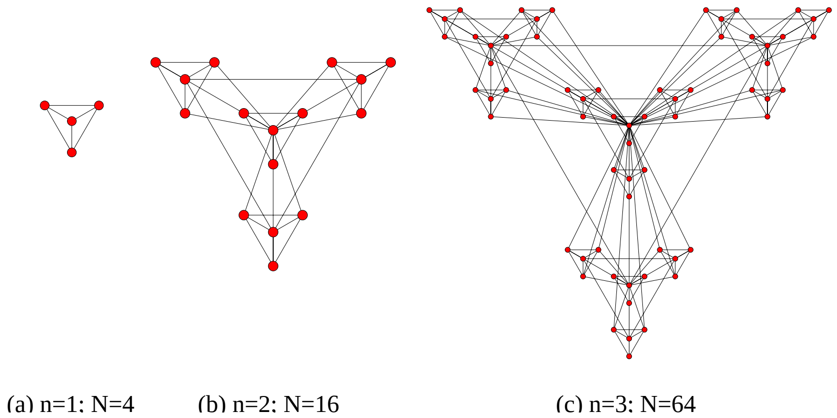


Figure S4. The construction of the hierarchical model. The three panels (a-c) correspond to the three steps of the construction process.

Our starting point is a small cluster of four densely linked nodes (Fig. S4a). Next we generate three replicas of this hypothetical module and connect the three external nodes of the replicated clusters to the central node of the old cluster, obtaining a large 16-node module (Fig. S4b). At the subsequent step we again generate three replicas of the obtained 16-node module, and connect the peripheral nodes to the central node of the old module (Fig. S4c). These replication steps can be repeated indefinitely, in each step quadrupling the number of nodes in the system. The emerging architecture seamlessly integrates a scale-free topology with an inherent modular structure.

A key feature of the obtained network, not shared by either the scale-free (Fig. 1a, article) or modular (Fig. 1b, article) models, is its hierarchical architecture. This hierarchy is evident from a visual inspection: the network is made of numerous small, highly integrated four node modules, which are assembled into larger 16-node modules, that are less integrated but each of which is clearly separated from the other 16-node modules. These in turn form 64-node modules, which are even less cohesive, but again will appear separable if the network expands further.

1.2.2 Properties of the hierarchical model

To analyze the scaling behavior of the degree distribution and clustering coefficient of this hierarchical network we first need to count the nodes with different degrees as well as their clustering coefficients. Starting with the first four nodes, we label the middle one a "hub" and we call the remaining three "peripheral". All nodes that originate as copies of hubs are again called "hubs", and we will continue calling copies of peripheral nodes peripheral. This distinction is useful since the rules responsible for connecting these classes of nodes are somewhat different.

Let us first focus on the hubs. The central hub acquires 3^n links during the n^{th} iteration, while its copies are linked to each-other. Let us call the central hub H_n, the three copies of this hub H_{n-1}. The 3*4 leftover centers of modules who's size is equal to the one of the network at the $(n-2)^{\text{th}}$ iteration are called H_{n-2}.

At the n^{th} iteration a hub H_i has all the links the central hub had after the i^{th} iteration, and two in addition from its two neighboring modules of identical size:

$$k_i(H_i) = 2 + \sum_{l=1}^{i} 3^l = 2 + 3\,\frac{3^i - 1}{2}.$$

For any $i < n$ the number of H_i modules, $N(H_i)$, is $3 \cdot 4^{n-1-i}$ (there are three for $i = n-1$, $3 \cdot 4$ for $i = n-2$; for $i = 1$ we have $3 \cdot 4^{n-2}$; 3/4-th of the copies of the original tour-node module). Since we have

$$N(H_i) = 3 \cdot 4^{n-1-i}$$

H_i-hubs of connectivity $k(H_i)$, we obtain $\ln N = c_1 - i \ln 4$ and $\ln k \simeq i \ln 3$. Thus we have $\ln N = c_1 - k\,\frac{\ln 4}{\ln 3}$, which corresponds to a power-law degree distribution (Fig. S5a)

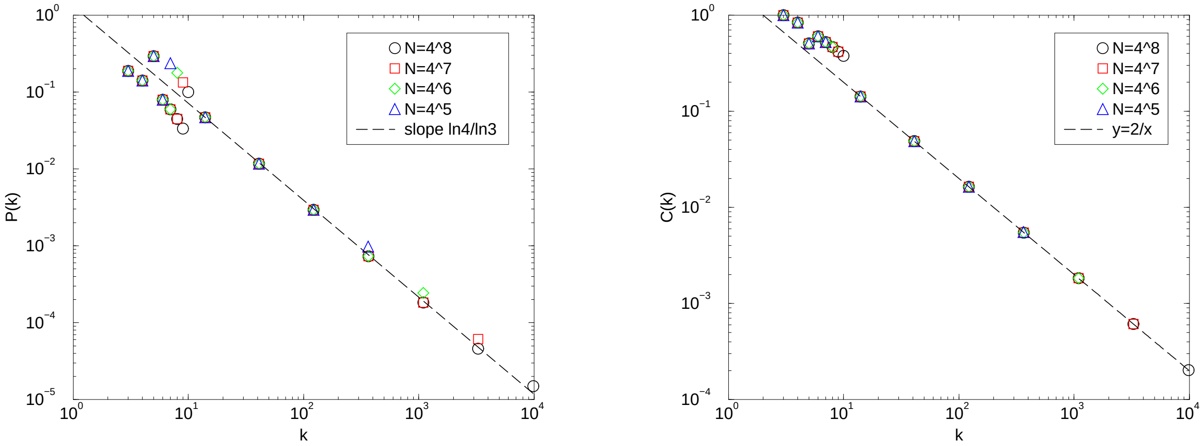
$$P(k) \sim k^{-\gamma}, \qquad \text{where} \qquad \gamma = 1 + \frac{\ln 4}{\ln 3}.$$

The clustering coefficient of the H_i hubs is easy to calculate. Their $\sum_{l=1}^{i} 3^l$ links come from nodes linked in triangles, thus the connections between them is equal to their number. There is one additional link between the two identical hubs H_i is linked to, so the number of links between the H_i hub's neighbors is $\sum_{l=1}^{i} 3^l + 1 = k_i - 1$. This gives

$$C(H_i) = \frac{k_i - 1}{k_i\,(k_i - 1)/2} = \frac{2}{k_i}, \qquad \text{or} \qquad C(k) = \frac{2}{k},$$

indicating that the $C(k)$ function for the hubs of this network scales as k^{-1} (Fig. S5b)[5].

Since the peripheral nodes can have maximum $n + 2$ links in a network of $N = 4^n$ nodes with maximum number of links of the order of 3^n, the scaling behavior of the $P(k)$ degree distribution, as well as the $C(k)$ function, is determined by the hubs. In Fig. S5 we plot $P(k)$ and $C(k)$ for computer generated model networks of different sizes.



(a) Degree distribution.

(b) Clustering coefficient.

Figure S5. The degree distribution $P(k)$ and the clustering function $C(k)$ of the hierarchical networks of different sizes (N).

The average clustering coefficient of the hierarchical model asymptotically approaches 0.606, the correction to its asymptotic value decreasing as a power law with the system size (Fig. S6).

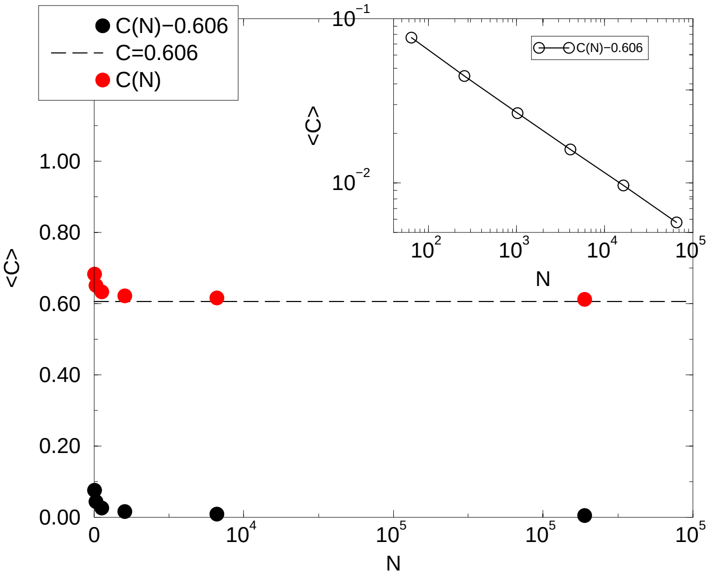


Figure S6. The size dependence of the average clustering coefficient of the hierarchical network. The inset indicates that the average clustering coefficient converges to its asymptotic value following $C(N) = 0.606 + 0.354 * N^{-0.372}$.

2. Graph Theoretic Representation of the Metabolic Network

2.1 Definition of the metabolic network

Substrates represent the nodes of a metabolic network while links represent the chemical reactions the substrates participate in. Our analysis requires an undirected graph, which we obtain by linking all in-coming substrates (educts) of a reaction to all its outgoing substrates (products) [6] (Fig. S7).

The *E. coli* metabolic network defined this way has $N = 885$ nodes, and it can be visualized using a standard clustering algorithm built into the Pajek graph drawing software (Fig. S8.)

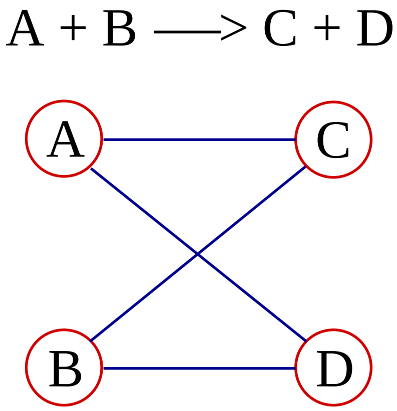


Figure S7. Graph theoretic representation of a reaction in a metabolic network.

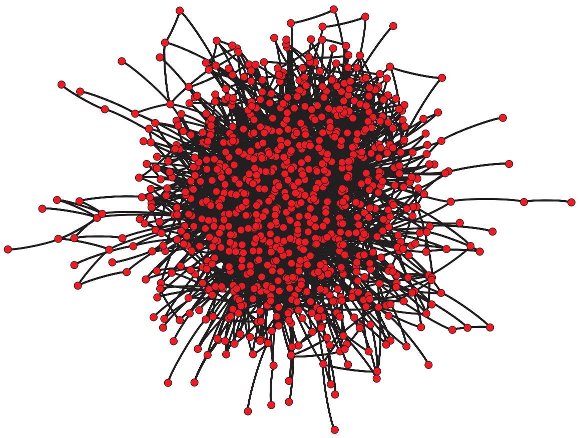


Figure S8. The complete *E. coli* metabolic network.

2.2 Generating the Reduced E. coli metabolic network

2.2.1 Biochemical reduction

In order to uncover the functional grouping of metabolites we need to take into account significant topological redundancies present in the simple graph theoretic representation of the metabolism. For example, a link from ATP, ADP, water, etc. to a metabolite A carries little biologically relevant information about the function of the metabolite A. ATP and ADP are usually responsible only for the energy exchange in most reactions. In addition, there are many different reactions where other pairs of metabolites help some reactions to take place (exchange of a proton or a phosphate moiety, for example), playing similar role to ATP or ADP.

In order to focus on biologically relevant transformations of substrates, we have performed a biochemical reduction of the metabolic network. Our guiding principle was to maintain on each pathway the main line of substrate transformation. In Fig. S9. we illustrate the reduction process, showing an original pathway map (left), the network corresponding to it (middle), and the network obtained after the reduction process (right).

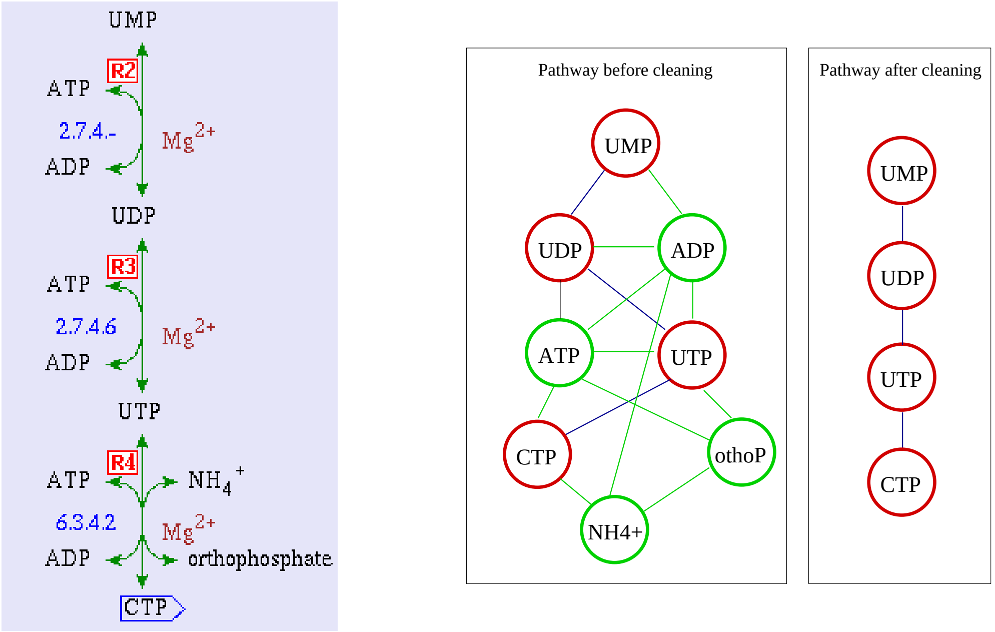


Figure S9. Biochemical reduction of the pathways of the metabolic network. The middle panel shows the full graph theoretic representation of the pathway shown in the left panel. The right panel displays the pathway after biochemical reduction.

It is important to note that the reduction process is completely local, i.e., it takes place at the level of each reaction, and does not result in the removal of metabolites, but only in the removal of links from the graph representation.

The resulting biochemically reduced metabolic network for *E. coli* is shown in Fig. S10.

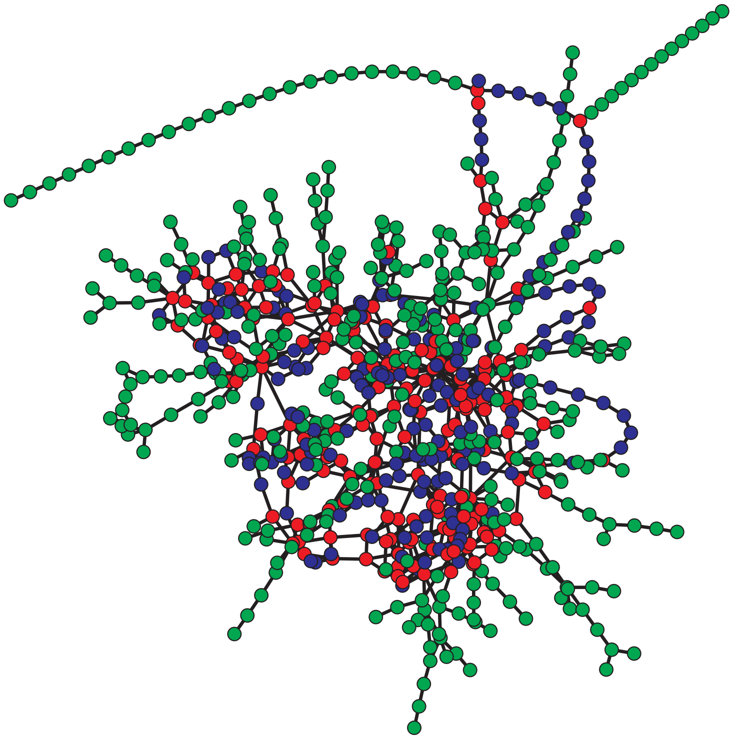


Figure S10. The *E. coli* metabolic network after biochemical reduction.

2.2.2 Topological reduction

To further reduce the complexity of the metabolism we continued the reduction process with a two-step topological reduction. As Fig. S10 shows, many pathways uncovered by the first reduction are connected to the rest of the metabolic network by a single substrate (green), or represent a long chain of consecutive substrates that appear as an arc between two substrates and have no other side-connections (blue). Since the topological location of the strings of substrates depend only on one or two multiply connected terminal substrates (red), we can temporarily remove the elements of the long non-branching pathways without altering the topology of the core metabolism.

We define as “hairs” (green) all sets of nodes that can be separated from the network by cutting one link. An “arc” (blue) is an array of nodes connected by only two links to the rest of the metabolism, leading from one well-connected substrate to another (red). To generate the reduced metabolic network we have removed all hairs from the network and replaced all arcs with a single link connecting directly the substrates at the two ends of an arc.

The algorithm for ”hair” and ”arc” reduction works as follows:

- Set the color of all nodes black.
- Find and color green all ”simple hair” nodes: start from a node with only one link and move along the links until a node with at least three links is encountered. Color green all nodes with two links encountered during this process. Repeat the above procedure starting from all nodes with only one link.
- Find and color all ”arcs” blue: all nodes which are not green but have only two links are colored blue.
- Find ”branching” hair nodes and color them green (for example, the fork-shaped hair on the right bottom of Fig. S10. Many of the nodes on this hair are blue at this stage): Start from a blue node and follow all links to non-green nodes. After finding the first blue-blue or blue-black link
    - Cut the blue-blue or blue-black link.
    - Starting from one end of the removed link perform a simple burning algorithm[1] to find out if the removal of this link separates a ”hair”.
        - If if *did not* separate a ”hair”: the found component is equal to the network’s giant cluster. No coloring in this case.
        - If if *did* separate a ”hair”: the found component is smaller than the network’s giant cluster. Thus, we have either burned part of a ”branching hair” or the rest of the network.
            - If the reached component is smaller than half of the network’s giant cluster, then call it a ”hair” and color all burned nodes green.

[1] Burning algorithm: Label all nodes −1 except the starting node. Label this node 0. Find all the nodes it’s linked to and label them 1. Then find all the nodes not yet labeled and linked to any node called 1 and label them 2. Then find all the nodes unlabeled and linked to label 2 nodes and call them 3. Continue until you do not find any more nodes. All nodes with non-negative labels are part of the network component that the starting node belongs to.

- · If the reached component is larger than half of the network's giant cluster, then burn from the other end of the removed link and color all burned nodes green.

  Repeat this procedure for all blue (or still blue) nodes.

- Color all remaining black nodes red.

- Create a new, half-reduced network by removing all green nodes and storing them as separate small networks, together with the label of the node they are attached to (For *E. coli* see Fig. S12(a)).

- Create a new, reduced network by removing all blue arcs and connecting their two red ends by a simple link. Again, store the arcs as separate small networks together with the two ends they connect to. (For *E. coli* see Fig. S12(b), recolored).

A schematic topological reduction can be seen on Fig. S11.

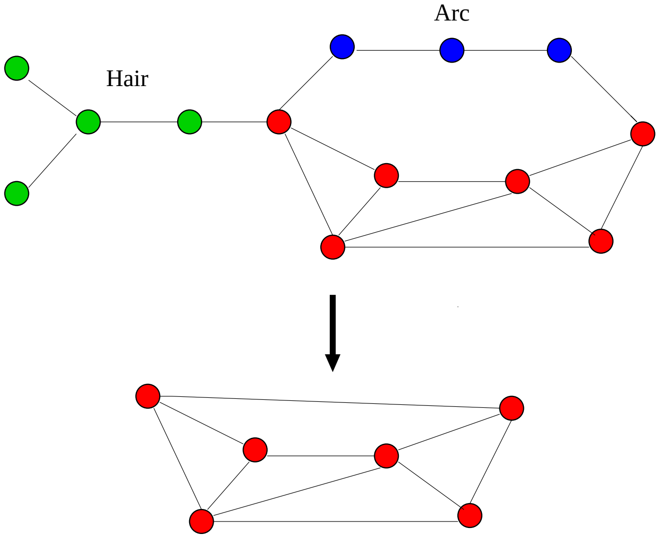
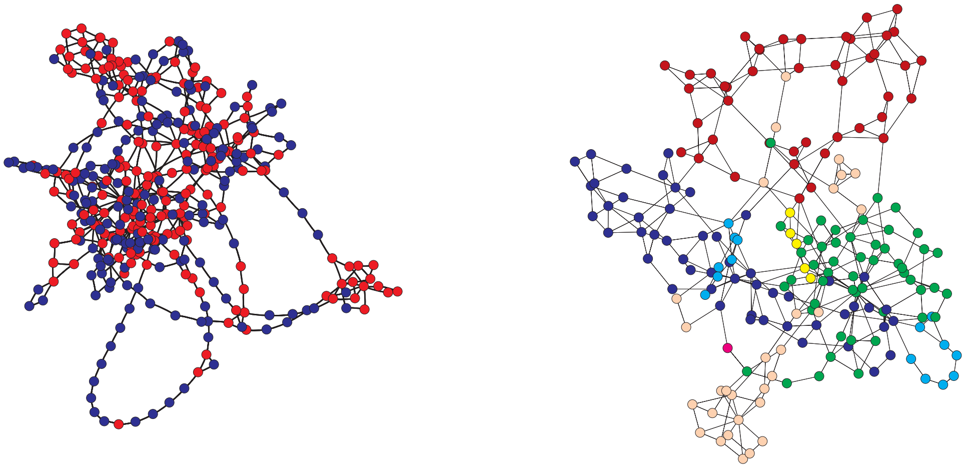


Figure S11. Topological reduction, which implies temporally removing all "hairs" (green) and replacing each "arc" (blue) with a single link.

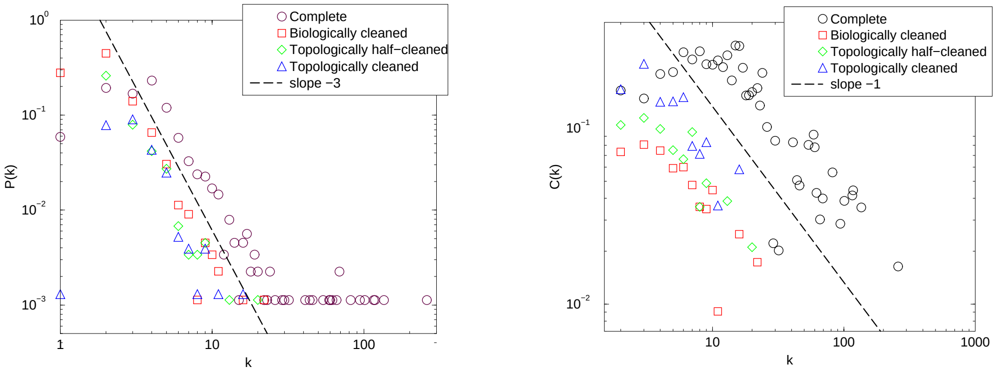
Note that we do not repeat the above described process on the reduced network. Thus, after the reduced network is ready, it can have arcs and hairs in it (see Fig. S12(b), light blue arc in right bottom corner). These appear, for example, when two linked "red" nodes both have hair on them, so they both have three links. After the reduction they are left with two links and form a newly created arc.

While the substrates removed during the topological reduction process are biologically important components of the network, their existence does not affect the subunit's connections to other parts of the metabolism. In this sense, they are topologically irrelevant. Note, however, that all removed substrates are re-added for the final biological analyses (Fig. 4d, article). The result of this two step reduction process for the *E. coli* metabolism is shown on Fig. S12.



(a) No hair. **(b)** No hair, shortened arcs.

Figure S12. Topological reduction of the metabolic network. Starting from the biochemically reduced metabolic network shown in Fig. S10, we removed all "hairs" (a) and "arcs" (b) from the network. The color code of the nodes in the final figure denotes the corresponding substrate's functional role (See Fig. 4b, article).



(a) $P(k)$. **(b)** $C(k)$.

Figure S13. Statistical properties of the reduced networks.

Both the biological and topological reduction process affects the connectivity and the clustering coefficient of the nodes, so it is important to note that these processes do not change the large-scale properties of the metabolic network. Fig. S13 shows the degree distribution and the clustering coefficient of the metabolic network obtained in different reduction stages. As the figure shows, the scaling of $P(k)$ and $C(k)$ remains largely unchanged during this process. This is not unexpected, as the reduction is purely a local process, which does not alter the networks's large scale features.

3. Clustering and Functional Characterization

3.1 The Overlap Matrix

The links between substrates of the reduced metabolic network can be used to define a topological overlap for all pairs of nodes, reflecting their interconnectedness. Using $l_{i,j} = l_{j,i} = 1$ if i and j are linked (0 otherwise) and the step-function Θ, the elements of the overlap matrix are given by

$$O_T(i,j) = \frac{{\sum_{l=1;}^{N}}' l_{i,l} \cdot l_{j,l} + \Theta(l_{i,j})}{\min(k_i, k_j) + 1 - \Theta(l_{i,j})}.$$

This measure of relatedness will have a high value if the nodes are linked to the same nodes (explaining the sum of all common neighbors and the division by the number of the smaller of the connectivity values), and will have the value $O_T = 1$ for linked pairs only.

The overlap matrix for the *E. coli* metabolic network, with alphabetically ordered substrates, is shown on Fig. S14. There is some grouping of the overlap values of nearby nodes, due to the fact that similarly named metabolites often have related functions.

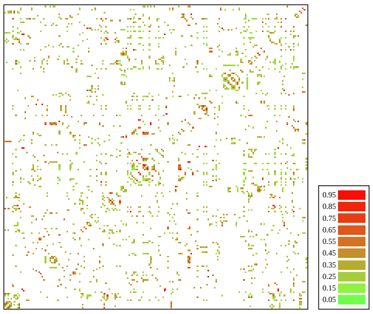


Figure S14. Overlap matrix for alphabetical ordering of the substrates in the *E. coli* metabolism.

3.2 Hierarchical Clustering

Our hierarchical clustering based on the overlap matrix uses the unweighted average linkage clustering algorithm, also known as UMPGA [7, 8]. The method finds the largest overlap present in the matrix, joins the corresponding substrates u and v to a branching point on the tree, and substitutes them with a "new" cluster $\{u, v\}$. This new item in the overlap matrix has the overlap with an arbitrary substrate (cluster) w

$$O(\{u, v\}, w) = \frac{n_u \cdot O(u, w) + n_v \cdot O(v, w)}{n_u + n_v},$$

where n_u is the number of components in cluster u. This definition ensures that all original overlap values are represented in the joint cluster's overlap value with the same weight, explaining the method's name as "unweighted average linkage clustering". The repetition of the above rules eventually shrinks the overlap matrix to a single value, corresponding to the root of the hierarchical tree, thus producing a tree with all the original substrates as its leafs, grouped naturally on branches reflecting their hierarchical overlap.

When the overlap values between clusters is redundant (i.e. there are at least two groups of clusters with the same overlap value) the program automatically joins the pair located first. The ordering of two branches under a junction is irrelevant, thus arbitrary. The distance between two junction levels is defined to be one.

The clustering of the *E. coli* metabolic network, and thus ordering the overlap matrix according to a substrate's horizontal location on the tree, lead to Fig. 4a (article).

3.3 Embedded Modularity of E. Coli Metabolism Based on its Topological Organisation

At the highest level we find that the E. coli metabolic network is partitioned into three large classes, appearing as major branches on the tree (Fig. S15 and Fig. 4a from article). The smallest of these branches consists of the Coenzyme and Vitamin metabolism (light orange), its inner core is divided into Vitamin K- and Terpene metabolism, while its outer part is specific to Sirohem Anabolism.

The second major branch represents the nucleotide and nucleic acid metabolism (red). It's two major sub-branches are clearly divided into the Pyrimidine and Purine metabolism subclasses. Interestingly, the purine group has a small sub-branch representing Dihydrofolate Anabolism, a subgroup that is shared with the Coenzyme and Vitamin metabolism (light orange) containing metabolites reacting within the subgroup. The strong link to Purine metabolism is due to the dihydroneopterin-triphosphate synthesis pathway from GTP.

The third and largest branch naturally breaks into a smaller branch containing largely Carbohydrate metabolism (blue), with most of the poly- and disaccharides in the branch on the left; while Monosaccharides- (some of which are also present in the left branch), Sugar Alcohols, and Alcohol metabolites dominate the right branch. The Membrane Lipid metabolism (cyan),(which is fairly independent of the Fatty Acid group) is nested into the Carbohydrates group due to shared glycerol metabolism susbstrates in its biosynthesis pathways. A small group representing Pyridoxine Anabolism (Vitamins: Vitamin 6B, light orange) is linked into this branch via biosynthesis from D-erythrose-4P. Another small nested group is the 3-phosphoshikimate biosynthesis from D-erythrose-4P, a part of Chorismate metabolism shared by both the Aromatic Compounds Metabolism (dark pink), and the Coenzyme group. The second major sub-branch is the least segregated one, as in addition to Proteins, Peptides and Amino Acids group (PPA, green) it contains several apparently unrelated pathways. One clear and separate sub-branch at the left side represents Fatty Acid metabolism, it being strongly linked to the Organic Acids and the Citrate Cycle (Carbohydrates, blue). Since almost half of the Amino Acid class substrates are shared with Carbohydrates Metabolism, pathways belonging to Pyruvate, Glyoxylate and Metabolism Sugar Alcohols are naturally grouped within the PPA group, appearing as small red branches on the figure. Formate metabolism, which represents almost the complete Monocarbon Compounds Metabolism class (yellow) is linked to Metabolism Sugar Alcohols. The IMP anabolic pathway (part of Purine metabolism, blue) starts with 5-phospho-'alpha'-D-ribose-1-diphosphate and the substrates on this pathway diverge from Purine metabolism, and are grouped on the PPA branch. Similarly, Nicotinamide metabolism (Coenzymes, light orange) is grouped into the PPA branch due to NAD(+) biosynthesis from L-aspartate. Enterobactin biosynthesis from chorismate pathway links parts of Chorismate metabolism (Coenzymes, light orange) to L-serine, the small (2 substrate) insert next to the Pyruvate group (a small blue group, shared by Carbohydrates and PPA). The pathway leading from L-Glutamate to L-glutamate-1-semialdehyde is part of Lipid, Aromatic Compounds and Coenzymes metabolism, its links anchor it into Glutamate metabolism. The PPA substrates on this large branch tend to group according to classifications based on the names of the amino acids, but not all of them show up on distinguishable sub-branches. They tend to group internally as well, for example most of glutamate and arginine metabolism substrates can be found on the same branch.

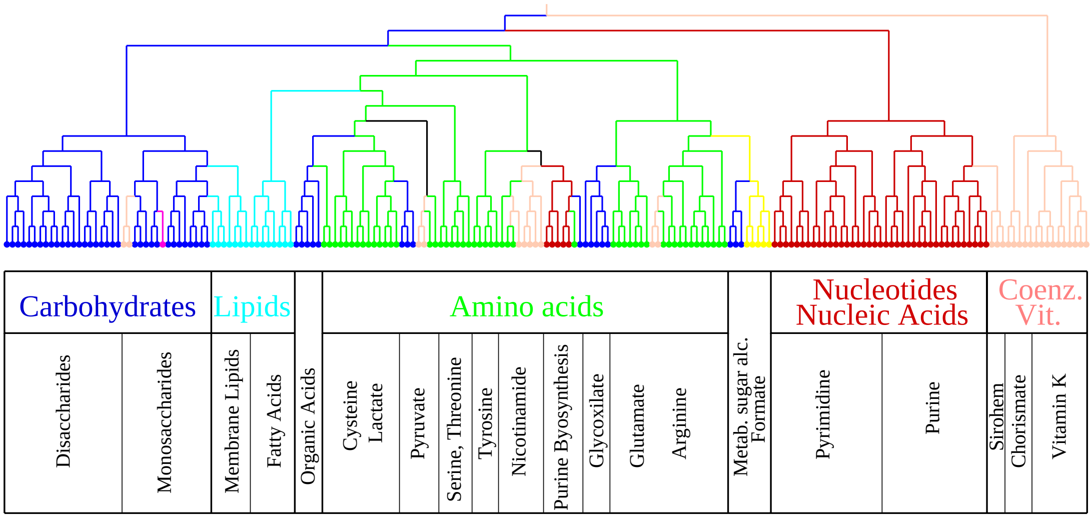


Figure S15. Hierarchical tree representing the reduced *E. Coli* metabolic network.

References

[1] A.-L. Barabási, R. Albert, *Science* **286**, 509-12 (1999).

[2] B. Bollobás, *Random Graphs*, Academic, London (1985).

[3] M. Girvan, M. E. J. Newman, Community structure in social and biological networks, Los Alamos Archive *cond-mat/0112110* (2001).

[4] D. J. Watts, S. H. Strogatz, *Nature* **393**, 440-2 (1998).

[5] S. N. Dorogovtsev, A. V. Goltsev, J. F. F. Mendez, Pseudofractal Scale-free Web, Los Alamos Archive *cond-mat/0112143* (2001).

[6] J. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, A.-L. Barabási, *Nature* **407**, 651-654 (2000).

[7] J. Sokal, P. Sneath, *Numerical Taxonomy*, Freeman, San Francisco (1973).

[8] M. B. Eisen, P. T. Spellman, P. O. Brown, D. Botstein, *Proc Natl Acad Sci USA* **95**, 12863-8 (1998)