BIOINFORMATICS

Vol. 17 no. 12 2001
Pages 1198–1208

Visualizing plant metabolomic correlation networks using clique–metabolite matrices

Frank Kose [1,*], Wolfram Weckwerth [1], Thomas Linke [2] and Oliver Fiehn [1]

[1]Max Planck Institute of Molecular Plant Physiology, Department of Lothar Willmitzer, Postfach, 14424 Potsdam, Germany and [2]University of Potsdam, Faculty of Informatics, AG Torsten Schaub, 14424 Potsdam, Germany

Received on September 1, 2000; revised on March 6, 2001 and May 24, 2001; accepted on June 26, 2001

ABSTRACT

Motivation: Today, metabolite levels in biological samples can be determined using multiparallel, fast, and precise metabolomic approaches. Correlations between the levels of various metabolites can be searched to gain information about metabolic links. Such correlations are the net result of direct enzymatic conversions and of indirect cellular regulation over transcriptional or biochemical processes. In order to visualize metabolic networks derived from correlation lists graphically, each metabolite pair may be represented as vertices connected by an edge. However, graph complexity rapidly increases with the number of edges and vertices. To gain structural information from metabolite correlation networks, improvements in clarity are needed.

Results: To achieve this clarity, three algorithms are combined. First, a list of linear metabolite correlations is generated that can be regarded as a set of pairs of edges (or as 2-cliques). Next, a branch-and-bound algorithm was developed to find all maximal cliques by combining submaximal cliques. Due to a clique assignment procedure, the generation of unnecessary submaximal cliques is avoided in order to maintain high efficiency. Differences and similarities to the Bron–Kerbosch algorithm are pointed out. Lastly, metabolite correlation networks are visualized by clique–metabolite matrices that are sorted to minimize the length of lines that connect different cliques and metabolites. Examples of biochemical hypotheses are given that can be built from interpretation of such clique matrices.

Availability: The algorithms are implemented in Visual Basic and can be downloaded from our web site along with a test data set (http://www.mpimp-golm.mpg.de/fiehn/projekte/data-mining-e.html).

Contact: kose@mpimp-golm.mpg.de

*To whom correspondence should be addressed.

INTRODUCTION

In the post-genomic era, gene function elucidation will be the focus (Oliver, 1997). A number of different approaches to gain comprehensive data at every cellular level are being developed that relate gene mutations or stress conditions to changes in gene expression patterns at mRNA or protein levels (Fiehn *et al.*, 2001). Recently, we have extended the idea of profiling gene products to the metabolite level (Trethewey *et al.*, 1999) using gas chromatography coupled to mass spectrometry to cover mono- to trisaccharides, fatty acids and -alcohols, hydroxy- and amino acids, polyamines, sterols, alcohols, sugar alcohols, and miscellaneous compounds (Fiehn *et al.*, 2000a). It was shown that the analytical precision in determining metabolite levels (<5% RSD) was well below the biological variability (>30% RSD) (Fiehn *et al.*, 2000b). Metabolomic analysis by gas chromatography/mass spectrometry is complementary to approaches using infrared spectroscopy (Johnson *et al.*, 2000), NMR analysis (Gavaghan *et al.*, 2000), or two-dimensional thin-layer chromatography coupled to radioactivity detection (Tweeddale *et al.*, 1999). However, the use of chromatographic separation before multivariate spectrometric detection seems advantageous for quantifying and identifying as many individual compounds as possible in mixtures of thousands of metabolites. By applying metabolite profiling snapshots to comparisons of mutants and wild type plants, metabolite levels can be summarized to 'metabolic phenotypes' (Fiehn *et al.*, 2000b) by Principal Components Analysis (PCA). Multigene variations such as those existing between different wild type ecotypes of the same species reveal larger metabolic differences than single point mutation/parental wild type comparisons. This is true even when a single point mutation leads to a strong dwarf phenotype such as in the case of the *dgd1* mutant of *Arabidopsis thaliana*, which is impaired in the biosynthesis of the thylakoid membrane lipid DGD (Dörmann *et al.*, 1995).

Once PCA pattern recognition reveals different clusters, these clusters may be treated as individual populations for statistical analysis. However, during both pattern recognition and statistical analysis, information about metabolite regulation in the individual snapshots gets lost in the process of data reduction (e.g. averaging or linear combination). Therefore, methods are needed that utilize metabolite data in a more biochemically directed way. Recently, it has been proposed that metabolite data might be the key to elucidating novel gene functions (Teusink *et al.*, 1998). New algorithms based on stoichometric analysis of standard metabolic pathways have been shown to predict new pathways in cellular compartments (Pfeiffer *et al.*, 1999; Schuster *et al.*, 2000). Researchers focusing on metabolic control analysis emphasize the importance of profiling analyses in understanding the effects on metabolic networks when changing the activity of specific enzymes (Kell and Mendes, 2000). Mathematical modelling of metabolism might then direct plant bioengineering (Giersch, 2000; Gombert and Nielsen, 2000). Further algorithms have been developed that aim to engineer biochemical pathways in a more general way (Mendes and Kell, 1998). The ultimate goal of profiling techniques is to qualitatively and quantitatively define differences between different biological samples. The number of independent variables (mRNA, proteins, or metabolites) is only limited by technical advances and methods. Pair-wise comparisons among such variables are regularly found to represent important biological functions, such as protein–protein interactions, but also DNA–protein or DNA–metabolite interactions. In this report, metabolite–metabolite pairs are compared with the long-term objective to reveal novel regulation mechanisms or potential biochemical pathways. Once multiple interactions among variables can be screened experimentally by analytical tools, it seems adequate to visualize the resulting networks instead of giving the information in the form of tables. It will be important for any visualization of correlation networks to obtain immediate information about structural organization. Graphs currently used to visualize network interactions between specific biological variables (Schwikowski *et al.*, 2000), however, lose clarity if not restricted to a small number of vertices. One obvious way to improve clarity is to reduce the number of edges. Here we report a novel approach to compute correlations between metabolite levels, and to create a graph based on these correlations by structuring the visualization using maximal cliques.

ALGORITHMS

Metabolite correlations

In order to detect interactions between metabolites, metabolite levels of a series of physiological snapshots can be compared in pairs to test for their involvement in synchronous cellular processes. The coefficients, r_{xy} for linear correlations of pairs of metabolites that are calculated after normalization, regulate both members of a given pair. r_{xy} is defined as follows.

$$r_{xy} = \frac{m_{xy}}{s_{xy}} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2 \sum_{i=1}^{n}(y_i - \overline{y})^2}}. \quad (1)$$

These correlation coefficients can be used as estimates of how tightly the proposed metabolic links are controlled—the degree of control results in weaker or stronger fixation of metabolite–metabolite ratios. To allow researchers to define their own thresholds by which to define the cellular co-regulations of metabolite pairs, we made it possible to vary the threshold for r_{xy} according to user-defined criteria. After defining a correlation threshold, lists of metabolite correlations are generated from raw data. A further feature allows users to define a threshold for the minimal number of metabolite pairs necessary for calculating a correlation. Similarly, the empirical regression coefficients, b_{xy}

$$b_{xy} = r\frac{s_y}{s_x} = \frac{m_{xy}}{s_x^2} \quad (2)$$

give the slopes of the corresponding regression curves

$$y = \overline{y} + b_{y,x}(x - \overline{x}). \quad (3)$$

Regression slopes can then be used to determine whether ratios of correlating metabolite pairs are altered or not. These different slopes may indicate changes in metabolite fluxes across branching points in biochemical pathways.

An example from a metabolomic data set of over 184 polar metabolites from 45 individual *dgd1* mutant plants is given in Figure 1. Serine, isoleucine, and leucine show an obvious correlation to threonine levels with $r_{xy} > 0.90$ for each metabolite pair, although known biochemical pathways do not point to direct enzymatic conversions. Furthermore, isoleucine and leucine have almost identical regression slopes indicating that there is tight control over relative metabolite ratios along amino acid biosynthetic pathways. When compared to valine levels, leucine and isoleucine levels indicate that the metabolic control of relative metabolite ratios is occurring ($r_{xy} = 0.83$ and 0.89), but this control is clearly less pronounced for threonine ($r_{xy} = 0.70$), and is completely lost for serine ($r_{xy} = 0.59$). Linear correlations of $r_{xy} < 0.8$ can at best be interpreted as trends. Regularly, hundreds of metabolite correlations can be found with $r_{xy} > 0.80$. In this work, we aimed to develop tools to visualize each and

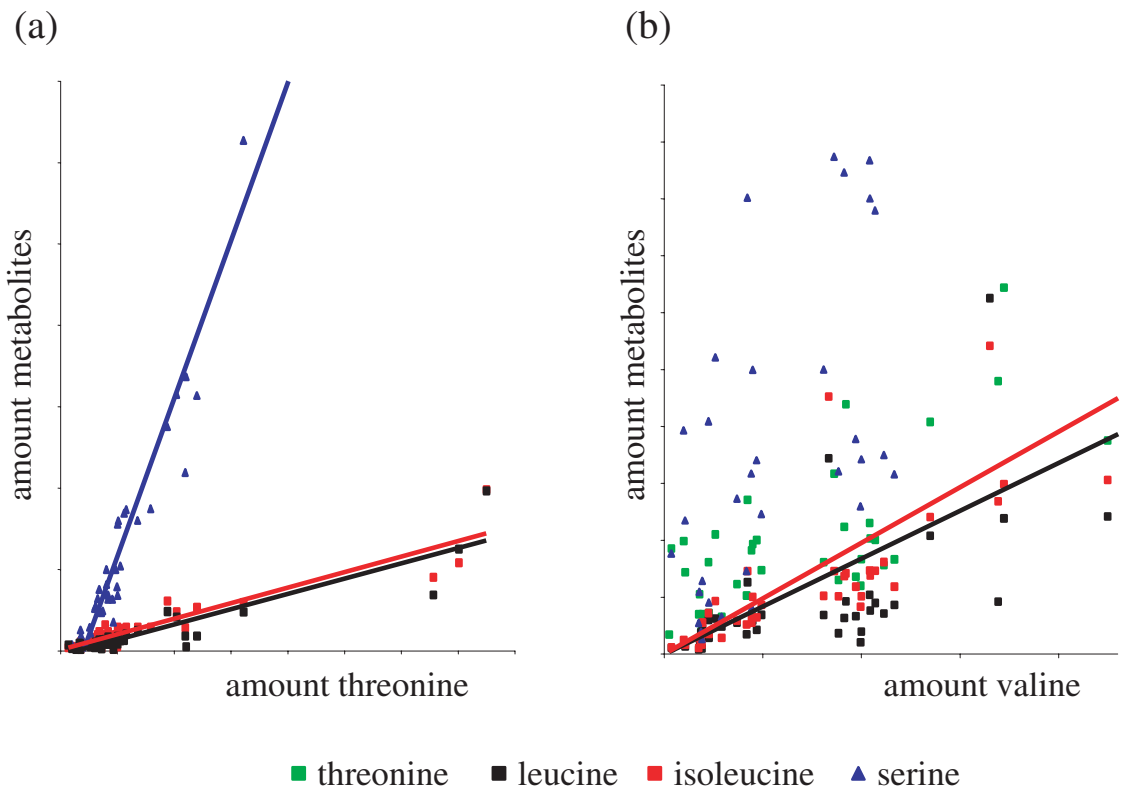

Fig. 1. Plots of metabolite/metabolite levels for individual plants. Correlations are shown by regression curves. (a) Serine, isoleucine, leucine versus threonine. (b) Threonine, leucine, isoleucine, serine versus valine. For threonine ($r_{xy} = 0.7$) and serine ($r_{xy} = 0.59$), correlation coefficients did not match the thresholds. Therefore, no correlation curve is plotted.

every correlation as part of a greater network in order to gain information about potential links and control points in plant metabolism. One way to visualize such complex graphs is to generate cliques derived from pairs of fully correlated metabolites.

The algorithm works to order all metabolites in a lexicographic way. Following this, every metabolite corresponding to the lexicographic sequence is proofed for correlations with all following metabolites. The result is a comprehensive list of correlations that consists of sublists with one identical metabolite in every correlation.

Maximal cliques of metabolite correlations

Graphs are regularly used to derive information from lists of variable correlations in a convenient way. A graph $G = (V, E)$ consists of a finite set of vertices V and a finite set of edges E. Elements of E have the form (xy) where $x, y \in V$ are vertices. In this paper, we consider only undirected graphs without loops ($(xx) \notin E$). Edges represent biological relationships, e.g. in our case, the linear correlation between pairs of metabolites, which are elements of V. If visualization tools are employed that try to symbolize all edges simultaneously, the resulting graph will lose clarity due to the increasing number and complexity of edges. We suggest to take the graph apart into maximal cliques, and, thus, to reduce graph complexity without losing information. To define maximal cliques, the term 'adjacency' is used.

DEFINITION. Two vertices x and y are adjacent, if they are connected via an edge (xy).

A graph $G' = (V', E')$ is called a subgraph of a graph $G = (V, E)$ if $V' \subseteq V$, $E' \subseteq E$ and each edge $(xy) \in E$ with $x, y \in V'$ is also an edge of G'. A graph $C = (V_C, E_C)$ is a clique in G, if C is a subgraph of the graph G such that for each two vertices $x, y \in V_C$ we have $(xy) \in E_C$. The set of vertices V_C is fully adjacent. A clique is also called a complete graph. A clique C, being a subset of G, is a maximal clique of G if there is no clique H such that C is a proper subset of H. Accordingly, we will call a clique with n vertices an n-clique. In order to distinguish cliques, we define:

$$C = \{(a \ldots n) | a, \ldots, n \in V_c\} \tag{4}$$

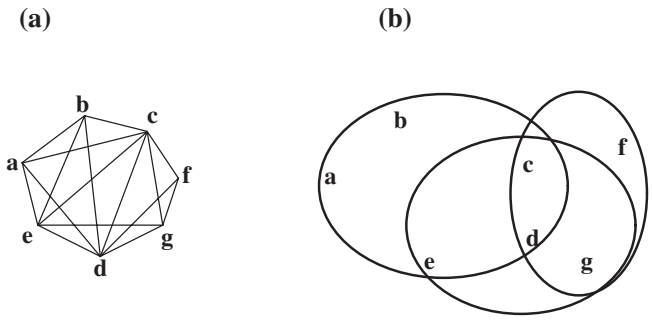

Fig. 2. Example of an undirected test graph. (a) Maximal cliques {abcde}, {cdeg} and {cdfg} are marked by clique covers. (b) Algorithms to find these maximal cliques are further explained in the text and in Figures 3 and 4.

with n giving the number of vertices that are combined in a clique. All vertices that are members of a clique are numbered from 1 to n. A list of all cliques including n vertices is indicated as C_n. For the description of the proposed algorithm, it is necessary to divide C_n according to identical vertices. If cliques in C_n have $(n-1)$ identical vertices, they are summarized to an $(n-1)$-sublist. As seen in the Section **Metabolite correlations**, all correlations are sorted in a list consisting of sublists with one identical metabolite in every correlation. All correlations can be regarded as 2-cliques with two vertices. Hence this list is identical with C_n consisting of $(n-1)$-sublists with $n = 2$. Additionally, we define $(n-2)$-sets, including all cliques from C_n with $(n-2)$ identical vertices.

The disassembly of a graph into maximal cliques is called clique cover. In Figure 2, a test graph G with $V = \{a, b, c, d, e, f, g\}$ is demonstrating the usefulness of using maximal cliques instead of drawing all edges for visualizing complex relationships. This is possible without losing information according to the definition of a clique, which consists only of fully adjacent vertices. Missing edges are much easier to be identified in Figure 2b than it is in Figure 2a. This is especially useful in cases when there are only some edges missing for a new maximal clique. In this way, a graph can be taken apart into maximal cliques, which may overlap. Overlapping areas of maximal cliques contain submaximal cliques.

Our algorithm is based on the definition that any subgraph consisting of two adjacent vertices (in our case, correlating metabolite pairs) can be regarded as a 2-clique. The task, therefore, is to test if these 2-cliques are maximal cliques. Any clique is submaximal if it can be combined with another clique to form a new, enlarged clique.

Two 2-cliques having one vertex in common can be combined to form a new 3-clique if there is an additional edge between the two sole vertices. Similarly, the combination of two 3-cliques having two vertices in common to form a new 4-clique can be tested. Two n-cliques need only be tested to form a new $(n+1)$-clique if both cliques have already at least $(n-1)$ vertices in common. Therefore, our algorithm starts with C_2 and looks for all possible combinations of 2-cliques inside of all $(n-1)$-sublists. The information indicating a possible edge between sole vertices is included in the corresponding $(n-2)$-set and identical with the $(n-1)$- and n-vertex. In the case of C_2, the $(n-2)$ vertex is empty and C_2 is identical to the $(n-2)$-set with $n = 2$. The result is a list of all 3-cliques, called C_3. This procedure is repeated as often as new cliques can be generated. All combined cliques are assigned by the $(n-2)$-vertex of the newly formed clique. The assignment process enables to mark submaximal cliques and to avoid the generation of unnecessary submaximal cliques: during the generation of all possible cliques from a $(n-1)$-sublist, it is possible that other $(n-1)$-sublists according to the same $(n-2)$-set can be fully assigned by the same vertex. Clique generation from such sublists can only lead to submaximal cliques and therefore need not be made. Once a clique is assigned, the assignment is not changed anymore. In order to distinguish assignments, we define:

$$C = \{(a \ldots n)_{\text{assignment}} | a, \ldots, n \in V_c\}. \qquad (5)$$

Using the test graph of Figure 2, the algorithm is exemplified in Figure 3, which also demonstrates the assignment of submaximal cliques. In Figure 3, all sets with $(n-2)$ identical vertices are divided by fat lines, while $(n-1)$-sublists are divided by thin lines. The vertices n and $(n-1)$ of $(n-2)$-sets contain the information of edges. For example, the cliques (abc) and (abd) are identical in the first $(n-1)$ vertices with $n = 3$, i.e. (ab). For a combination of these two cliques, there must be an edge (cd). This is identical to a constrained search for a clique (acd) in the corresponding $(n-2)$ set. Since this is the case, all three cliques (abc), (abd), and (acd) are simultaneously combined to generate the new 4-clique (abcd). All three combined cliques can now be assigned by the $(n-2)$ vertex of the new 4-clique, b.

Similarly, the cliques (abc), (abe), and (ace) are combined to a new 4-clique (abce) and are also assigned by the corresponding $(n-2)$-vertex, b. The last b assignment is carried out after combination of (abd), (abe), and (ade) to the 4-clique (abde). All 3-cliques in this $(n-2)$-set now have been assigned by b, and any further combination within the sublists of this set cannot lead to maximal cliques. As an example, a combination of (acd), (ace), and (ade) to (acde) is later found to be already included in the final maximal clique (abcde). This is true independent of clique sizes. Any further testing of fully assigned $(n-1)$-sublists can be stopped. However, if cliques in a sublist

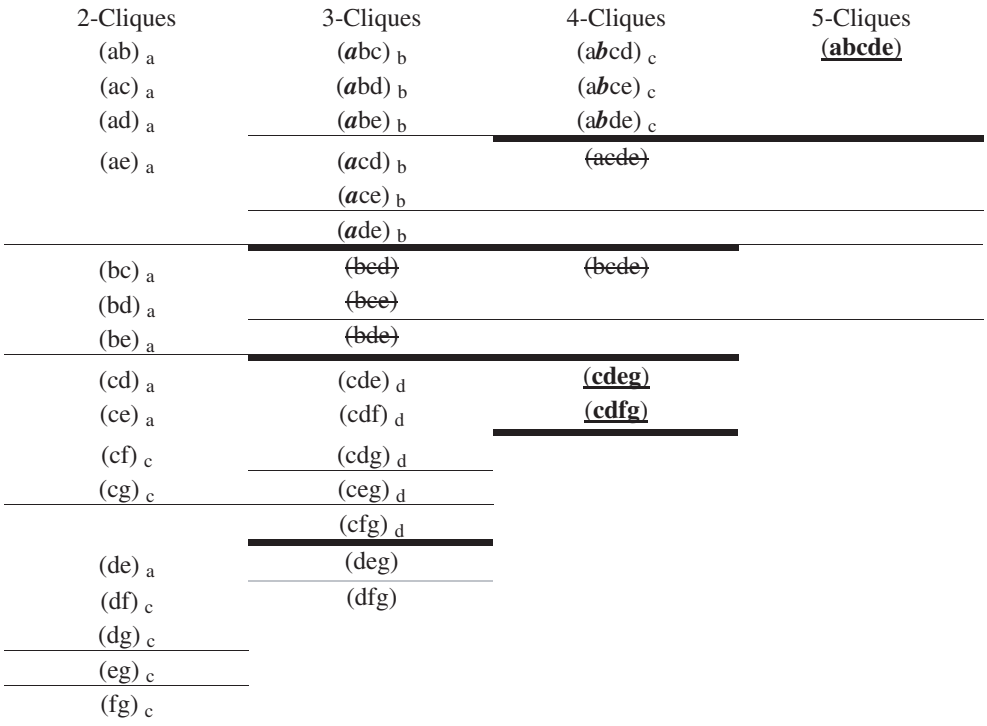

Fig. 3. Cliques that are generated by the proposed algorithm using the test graph in Figure 2. Fat lines divide lists of cliques into $(n-2)$ sets consisting of all cliques that have $(n-2)$-vertices in common. Thin lines divide lists into $(n-1)$-sublists. The branch-and-bound tree visualizes how the three maximal cliques in the example graph are found. Further, the assignment of cliques is exemplified. By combination of 2-cliques to generate the first sublist of 3-cliques (***a***bx), the 2-cliques are assigned by the index '$_{\mathrm{a}}$'. Correspondingly, the cliques of this 3-clique subset are assigned by '$_{\mathrm{b}}$' during combination to generate the next larger subset of 4-cliques (a***b***xy). When all cliques of a subset are assigned by an identical index, any further combination cannot lead to new maximal cliques. Cliques that were not generated by our algorithm are crossed out, such as (acde). Due to lexicographic ordering, the two cliques (dfg) and (deg) could not be eliminated by assignment but were later found as included in a maximal clique. For further explanations, see text.

are assigned differently, such as in $(\mathrm{cd})_a$, $(\mathrm{ce})_a$, (cf), and (cg), further combinations of these cliques have to be carried out. Subsequently, it leads to an assignment of (cf) and (cg) with c. Assigned cliques are submaximal, and in Figure 3, the final maximal cliques are marked in bold. For this simple test graph G with $V = \{\mathrm{a}, \mathrm{b}, \mathrm{c}, \mathrm{d}, \mathrm{e}, \mathrm{f}, \mathrm{g}\}$, a total of 40 cliques are found, 3 of which are maximal cliques. By the assigning procedure, five submaximal cliques were avoided to be generated using our algorithm. As a result of clique assignment, the number of non-generated cliques increases with increasing graph dimensions, as demonstrated in the Section **Application and discussion** below. Due to the lexicographic order of metabolite names, some submaximal cliques may not be eliminated along the branch-and-bound tree, such as (deg) and (dfg) for our test graph. If the vertices in our test graph were named differently, zero to four of such non-assigned submaximal cliques would have been found. These cliques are removed at the last step of the algorithm by testing to see if any non-assigned cliques are fully included in a larger clique. This assignment is a powerful tool to minimize the branch and bound tree as can be seen in Table 1.

In conclusion, our algorithm progresses in the following steps to find all maximal cliques:

(1) Start with all 2-cliques, that are sorted by lexicographic order.

(2) Divide all cliques into $(n-2)$-sets. (For 2-cliques, the only $(n-2)$-set is C_2)

(3) Subdivide these sets into $(n-1)$-sublists.

(4) Take a $(n-1)$-sublist and test if all cliques have identical assignments. If this is the case, go to the next $(n-1)$-sublist.

(5) If not, test all possible pairs of cliques for potential edges connecting the n-vertices. An edge is present, if there is a clique in the corresponding $(n-2)$-set that has an $(n-1)$- and an n-vertex identical to the n-vertices of two tested cliques.

Table 1. Number of maximal cliques found in a metabolomic data set of 184 metabolites and 45 individual plant samples. The graph dimension (and the resulting total number of cliques) was altered by varying the threshold for linear metabolite–metabolite correlations

Correlation threshold r	Total no. of maximal cliques	Total no. of cliques	No. of non-generated cliques due to the assignment	No. of submaximal cliques later found to be included in maximal cliques
0.98	6	9	0	0
0.97	14	17	0	0
0.96	18	28	0	0
0.95	26	54	6	0
0.94	29	73	6	0
0.93	29	105	11	1
0.92	33	146	21	5
0.91	41	301	67	4
0.9	45	589	170	11
0.89	57	896	327	16
0.88	67	1 511	680	21
0.87	70	1 990	956	24
0.86	86	3 584	2 068	32
0.85	95	5 435	3 303	57
0.84	89	11 169	7 320	63
0.83	102	17 828	11 815	125
0.82	108	29 800	20 399	188
0.81	130	57 052	42 028	395

(6) If there is an edge, combine all vertices of the three cliques to form a new $(n + 1)$-clique. Assign all combined cliques by the $(n - 2)$-vertex of the new $(n + 1)$-clique, if they have not already been assigned.

(7) Write a new list that contains all new $(n+1)$-cliques.

(8) Repeat this algorithm for all $(n - 1)$-sublists.

(9) Take the list generated in step 7 and repeat steps 2–8 as long as new cliques can be generated.

(10) At the end, all lists are tested, if any non-assigned n-clique is fully included in a $(n + x)$-clique. In this case, such n-cliques are eliminated.

Using this algorithm, all maximal cliques will be generated from arbitrary undirected graphs and sorted into lists.

Finding all maximal cliques in arbitrary, undirected graphs is an np-hard problem, for which an efficient branch-and-bound process was developed nearly 30 years ago (Bron and Kerbosch, 1973). The simple version of the Bron–Kerbosch algorithm generates each clique (maximal or submaximal) on the way along the branch and bound tree only once. It is based on the three sets *compsub*, *candidates*, and *not*. For a comparison with the Bron–Kerbosch algorithm, $(n - 1)$-sublists can be understood as a state of the sets *compsub* and *candidates*. All identical vertices in a $(n - 1)$-sublist correspond to the vertices in the set *compsub*, and the set of all different vertices in an $(n - 1)$-sublist corresponds to the vertices in the set *candidates*. A $(n - 1)$-sublist containing only one clique corresponds to an empty *candidate* set. All further possible extensions of the set *compsub* correspond to all possible combinations of cliques in a $(n - 1)$-sublist.

Since Bron and Kerbosch's contribution, numerous variants for finding maximum maximal cliques (Carraghan and Pardalos, 1990; Babel, 1991; Masuda *et al.*, 1990; Pardalos and Xue, 1994), and for detecting all maximal cliques of a graph (Osteen and Tou, 1973; Osteen, 1974; Balas and Toth, 1985; Gardiner *et al.*, 1997) have been reported. Several of these approaches are compared in a most recent fundamental study (Koch, 2001). Using these correlations, our algorithm provides a feature not available when using the Bron–Kerbosch algorithm: by combining cliques inside of lists and by introducing the assignment technique, our algorithm is a powerful tool to avoid non-maximal clique-generation.

Visualization

The third algorithm uses these lists in order to visualize structures in the resulting network of cliques and vertices. This network is visualized by a vertex–clique matrix that orders metabolites in columns and cliques in rows. The presence of a vertex in a clique is indicated by colouring its corresponding cell green. As further information, the number of cliques in which a vertex is found is given as number. Empty cells connecting different cliques are marked by vertical blue lines. Empty cells connecting different vertices of the same clique are marked by horizontal brown lines. In order to minimize the number of empty cells that connect cliques and vertices, rows and columns are ordered by the following algorithm:

(1) Search for the column with the largest number of green cells and set it to position p with $p = 1$.

(2) Search for the column that has the largest number of connections to all columns that have already been set and set this column to the position $p + 1$.

(3) Repeat step 2 as long as columns can be added.

(4) Repeat step 1–3 for all rows.

The general shape of such a matrix is a diagonal line of green cells (clique/vertex pairs). Cliques that do not have further connections to the rest of the network are mostly found isolated at the lower right corner of the matrix (see Figures 7 and 8a). The visualization of our test graph from Figure 1 is given in Figure 4.

APPLICATION AND DISCUSSION

To test our approach, the algorithms have been implemented by visual basic programming as macros running

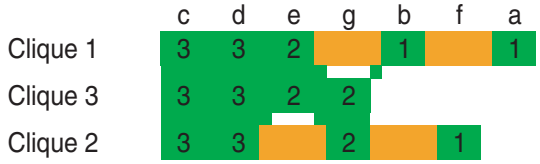

Fig. 4. Clique–vertex matrix visualization of the test graph from Figure 2. The presence of a vertex in a clique is indicated by colouring its corresponding cell green. As further information, the number of cliques in which a vertex is found is given as number. Empty cells connecting different cliques are marked by a vertical blue line. Empty cells connecting different vertices of the same clique are marked by a horizontal brown line.

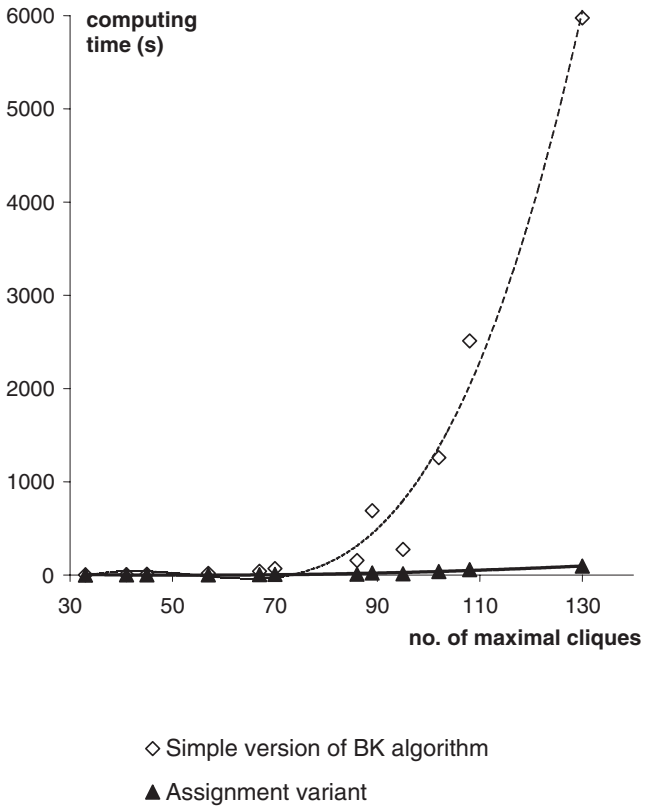

Fig. 5. Computing time due to number of maximal cliques in a graph based on a list of correlations generated from 184 metabolites and different correlation thresholds r.

on MS Excel 2000. Their performance was tested using a small data set of 45 metabolic snapshots covering 184 polar metabolites of the *A.thaliana dgd1* mutant, resulting in 8280 variables. Metabolite identities were confirmed by external and internal references (Fiehn *et al.*, 2000a). Metabolite levels were quantified relative to internal references and normalised to plant fresh weights (Fiehn *et al.*, 2000b). No further data scaling or pre-processing was applied. The number of cliques was increased by reducing the correlation threshold r_{xy} from 0.98 to 0.80. In Table 1, the total number of cliques within a graph is compared to the reduced number of cliques generated by the assignment process, and to the number of maximal cliques. This table clearly demonstrates that the power of the assignment process is increasing with increasing graph dimensions. Resulting, run times were compared using a PIII 600 MHz computer and 128 MB RAM. Figure 5 demonstrates that the proposed algorithm enabled largely reduced run times. For comparison, the run time is compared for the same data set to the original version of the Bron–Kerbosch algorithm, implemented in Visual Basic. Once all maximal cliques were found, stepwise clique rearrangement for clearest visualization took roughly 2 min. Figure 6 compares visualization by the freely available program DaVinci (Fröhlich, 1997) with the results using our algorithm. Without reducing graph complexity, it is essentially difficult to obtain any structural information. For detailed information on metabolite pairs, one must follow distinct edges. However, this is regularly not the focus of graph visualization since this information could also be obtained directly from correlation lists. In our approach (Figure 7), a matrix of cliques (in rows) and vertices (in columns) is used for direct visualization of relationships. All metabolites numbered by 1 are associated with only one clique and are isolated in the metabolic network with no direct connections to other cliques. Vice versa, metabolites that are numbered >1 are members of other maximal cliques. Some metabolites represent the only connection between two or more different cliques. Potentially, these connections could highlight branch points in biochemical pathways or routes bridging different metabolic cycles. Overlapping cliques are symbolized by vertically marked blue lines, whereas horizontally marked brown lines connect all metabolites of a clique. Detailed views of Figure 7 give further structural information. For example, a closed subgraph consisting of a 5-, a 4- and a 3-clique can be seen (marked as 'A'). Metabolites appearing in such a subgraph may occur only in certain compartments of plant cells. In the special case of this small example data set, an interpretation of an unconnected closed subgraph might also have the simple reason, that the missing vertices have not been analyzed. In addition, subgroups of metabolites can be identified that correlate to the same metabolites, based upon their presence in the same cliques. (e.g. U_ara38 and U_ara24 in Figure 8a). Metabolites in such subgroups show the same relationship in their correlations with all other metabolites. One interpretation of this relationship is that, these metabolites are synchronous to the remaining network. This structural information can easily be extracted from the network

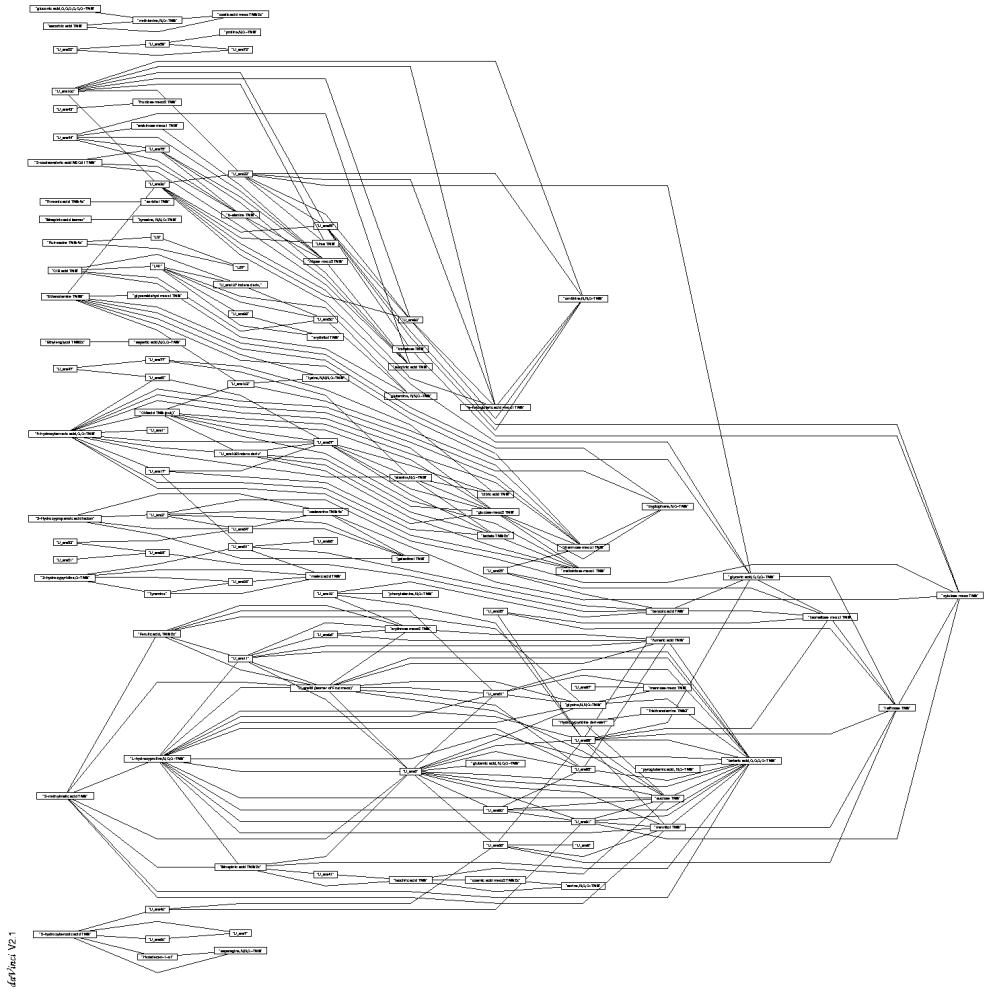

Fig. 6. Graph based on a correlation list generated from 184 metabolites. Vertices represent metabolites, edges represent correlations between metabolites.

independent of clique size. The algorithm described in the Section **Visualization** sorts these metabolites into spatial neighbourhood.

Another example in which the algorithm can be used to gain structural information is given in Figure 8b, which is a magnification of Figure 7 (marked there as 'B'). When examining the series of metabolites from U_13 to U_ara58, it can be seen that some metabolites, such as U_ara74, are not adjacent to other metabolites, such as U_ara58. However, these both metabolites are found to be connected by co-regulated metabolites, monopalmitin, U_ara78, and U_ara79. Again, we also find metabolites in this cluster that show identical relationships to all other metabolites (monopalmitin and U_ara79), which are indistinguishable, therefore. Such metabolites are supposedly very similar in their biochemical regulation.

This example demonstrates how to select missing edges from subgraphs. However, metabolite networks gained from linear correlations have to face limitations. A (linear) relationship between two metabolite levels, particularly a correlation, can be interpreted as a synchronized co-regulation of these metabolites. Since non-linear correlations are not yet covered, more general concepts to indicate relationships among expression data might be valuable such as mutual information (Shannon and Weaver, 1949; Baldi *et al.*, 2000) as well as clustering methods (Bittner *et al.*, 1999). For complex networks of more than 1000 variables, further visualization strategies have to be implemented to keep graphical clearness and to gain information from these graphs. Among these, a three-dimensional visualization of cliques is needed, including features such as visualizing the degree of overlap between different cliques and the strength of the

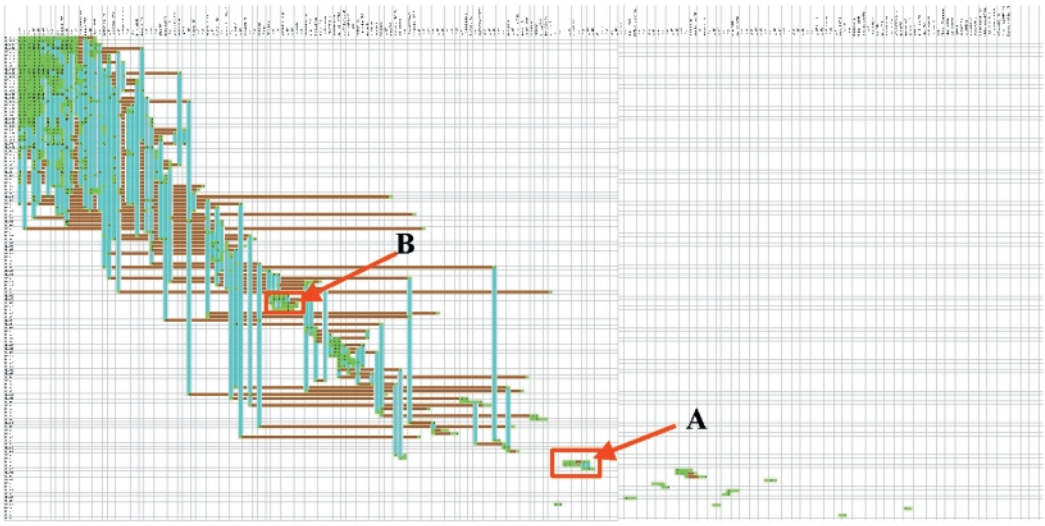

Fig. 7. The final graph is visualized by a vertex–clique matrix that orders metabolites in columns, and cliques in rows. The presence of vertex in a clique is indicated by colouring its corresponding cell green, with brown lines connecting all vertices of a clique, and blue lines connecting all cliques in which a particular vertex is member. Details A and B are explained in Section **Application and discussion** and Figures 8a and b.

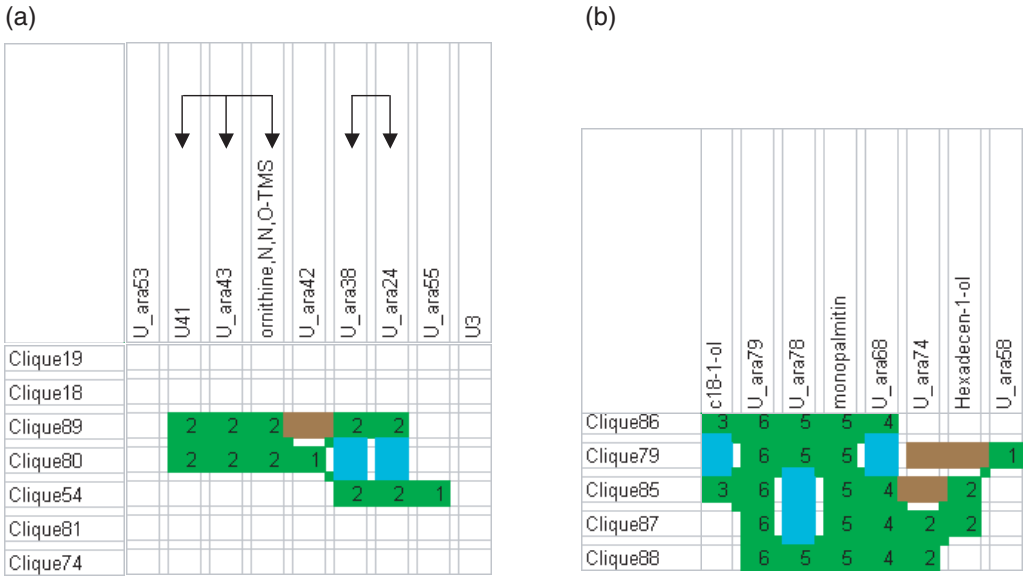

Fig. 8. (a) The enlarged detail A from Figure 7 shows an isolated graph. Arcs mark subgroups of metabolites revealing identical relationships to all other metabolites (both correlating and non-correlating). Such subgroups can be recognized by finding marked metabolites exclusively in the same cliques. (b) Enlarged detail B from Figure 7 shows a part of a graph. Due to missing horizontal connections missing correlations can be seen, e.g. The two metabolites U_ara74 and U_ara58 do not have connection in any clique, but are still found in the same cluster of cliques. Connections of both metabolites are found via monopalmitin, U_ara78, and U_ara79.

correlation factor r_{xy}. Further, comparisons of networks gained from different experiments (such as diseased and healthy tissues) have to be facilitated. In the next years it will become increasingly important to evaluate which chemometric and bioinformatic tools (or, combination of tools) can best be applied to gain biologically meaningful information from multivariate expression data. Obviously, this will strongly depend on the biological question to be

answered. For comparisons, data sets stored at publicly available sites might prove highly useful, as has been demonstrated recently for transcriptomic approaches (Gilbert *et al.*, 2000). It will prove useful to set up such a data bank for metabolomic analyses including accurate descriptions of all obtainable experimental and biological details.

CONCLUSIONS

The main point of analyzing correlations within metabolite profiling data sets is to obtain information about complex metabolite relationships. This has been achieved here by generating lists of maximal cliques that are sorted on matrices. These matrices provide a means to visualize overlapping cliques and their corresponding vertices in a compact way. Examples of structural information that can be derived through the analysis of such metabolite correlation networks include cliques with no further connections to the network, missing edges within overlapping cliques, and (connected) clusters of cliques. It has been shown that these algorithms can be applied to small data sets (max. 1024 metabolites) using widely available MS Excel programs. For larger data sets, 3D-visualization will be needed.

ACKNOWLEDGEMENTS

This project was funded by the Max-Planck-Society. We thank Megan McKenzie for editing the manuscript.

REFERENCES

Babel,L. (1991) Finding maximum cliques in arbitrary and in special graphs. *Computing*, **46**, 321–341.

Balas,E. and Toth,P. (1985) Branch and bound methods. In Lawler,E.L., Lenstra,J.K., Rinnooy Kan,A.H.G. and Shmoys,D.B. (eds), *The Travelling Salesman Problem*. Wiley, New York, pp. 361–465.

Baldi,P., Brunak,S., Chauvin,Y., Andersen,C.A.F. and Nielsen,H. (2000) Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, **16**, 412–424.

Bron,C. and Kerbosch,J. (1973) Finding all cliques of an undirected graph [H]. *Commun. ACM*, **16**, 575–577.

Bittner,M., Meltzer,P. and Trent,J. (1999) Data analysis and integration: of steps and arrows. *Nature Genet.*, **22**, 213–215.

Carraghan,R. and Pardalos,P.M. (1990) An exact algorithm for the maximum clique problem. *Oper. Res. Lett.*, **9**, 375–382.

Dörmann,P., Hoffmann-Benning,S., Balbo,I. and Benning,C. (1995) Isolation and characterization of an *Arabidopsis* mutant deficient in the thylakoid lipid digalactosyl diacylglycerol. *Plant Cell*, **7**, 1801–1810.

Fiehn,O., Kopka,J., Trethewey,R.N. and Willmitzer,L. (2000a) Identification of uncommon plant metabolites based on calculation of elemental compositions using gas chromatography and quadrupole mass spectrometry. *Anal. Chem.*, **72**, 3573–3580.

Fiehn,O., Kopka,J., Dörmann,P., Altmann,T., Trethewey,R.N. and Willmitzer,L. (2000b) Metabolite profiling for plant functional genomics. *Nature Biotechnol.*, **18**, 1157–1161.

Fiehn,O., Kloska,S. and Altmann,T. (2001) Integrated studies on plant biology using multiparallel techniques. *Curr. Opin. Biotechnol.*, **12**, 82–86.

Fröhlich,M. (1997) *Incremental Graph Layout in the Visualisation System daVinci*, PhD Thesis (in German), Department of Computer Science, University of Bremen, Germany, (http://www.informatik.uni-bremen.de/~davinci).

Gavaghan,C.L., Holmes,E., Lenz,E., Wilson,I.D. and Nicholson,J.K. (2000) An NMR-based metabonomic approach to investigate the biochemical consequences of genetic strain differences: application to the C57BL10J and Alpk:ApfCD mouse. *FEBS Lett.*, **484**, 169–174.

Gardiner,E.J., Artymiuk,P.J. and Willett,P. (1997) Clique-detection algorithms for matching three-dimensional molecular structures. *J. Mol. Graph. Model*, **15**, 245–253.

Giersch,C. (2000) Mathematical modelling of metabolism. *Curr. Opin. Plant Biol.*, **3**, 249–253.

Gilbert,R.J., Rowland,J.J. and Kell,D.B. (2000) Genomic computing: explanatory modelling for functional genomics. In Whitley,D., Goldberg,D., Cantú-Paz,E., Spector,L., Parmee,I. and Beyer,H.-G. (eds), *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2000)*. Morgan Kaufman, San Francisco, pp. 551–557.

Gombert,A.K. and Nielsen,J. (2000) Mathematical modelling of metabolism. *Curr. Opin. Plant Biol.*, **11**, 180–186.

Johnson,H.E., Gilbert,R.J., Winson,M.K., Goodacre,R., Smith,A.R., Rowland,J.J., Hall,M.A. and Kell,D.B. (2000) Explanatory analysis of the metabolome using genetic programming of simple, interpretable rules. *Genet. Progr. Evolv. Mach.*, **1**, 243–258.

Kell,D.B. and Mendes,P. (2000) Snapshots of systems. Metabolic control analysis and biotechnology in the post-genomic era. In Cornish-Bowden,A.J. and Cárdenas,M.L. (eds), *Technological and Medical Implications of Metabolic Control Analysis*. Kluwer Academic, Dordrecht, pp. 3–25.

Koch,I. (2001) Enumerating all connected maximal common subgraphs in two graphs. *Theor. Comput. Sci.*, **250**, 1–30.

Masuda,S., Nakajima,K., Kashiwabara,T. and Fujisawa,T. (1990) Efficient algorithms for finding maximum cliques of an overlap graph. *Networks*, **20**, 157–171.

Mendes,P. and Kell,D.B. (1998) Non-linear optimisation of biochemical pathways: applications to metabolic engineering and parameter estimation. *Bioinformatics*, **14**, 869–883.

Oliver,S.G. (1997) Yeast as a navigational aid in genome analysis. *Microbiol. UK*, **143**, 1483–1487.

Osteen,R.E. (1974) Clique detection algorithms based on line addition and line removal. *SIAM J. Appl. Math.*, **26**, 126–135.

Osteen,R.E. and Tou,J.T. (1973) A clique-detection algorithm based on neighborhoods in graphs. *Int. J. Comput. Inf. Sci.*, **2**, 257–268.

Pardalos,P.M. and Xue,J. (1994) The maximum clique problem. *J. Global Optim.*, **4**, 301–328.

Pfeiffer,T., Sánchez-Valdenebro,I., Nuno,J.C., Montero,F. and Schuster,S. (1999) METATOOL: for studying metabolic networks. *Bioinformatics*, **15**, 251–257.

Shannon,C.E. and Weaver,W. (1949) *The Mathematical Theory of Communication*. University of Illinois Press, Urbana.

Schuster,S., Fell,D.A. and Dandekar,T. (2000) A general definition of metabolic pathways useful for systemic organization and

analysis of complex metabolic networks. *Nature Biotechnol.*, **18**, 326–332.

Schwikowski,B., Uetz,P. and Fields,S. (2000) A network of protein–protein interactions in yeast. *Nature Biotechnol.*, **18**, 1257–1261.

Teusink,B., Baganz,F., Westerhoff,H.V. and Oliver,S.G. (1998) Metabolic control analysis as a tool in the elucidation of the function of novel genes. In *Methods in Microbiology*, Vol. 26, Academic Press, New York, pp. 297–336.

Trethewey,R.N,, Krotzky,A.J. and Willmitzer,L. (1999) Metabolic profiling: a Rosetta stone for genomics? *Curr. Opin. Plant Biol.*, **2**, 83–85.

Tweeddale,H., Notley-McRobb,L. and Ferenci,T. (1999) Effect of slow growth on metabolism of *Escherichia coli*, as revealed by global metabolite pool (‘metabolome’) analysis. *J. Bacteriol.*, **180**, 5109–5116.