BOX 1 KEY EXPERIMENTAL METHODS FOR SYSTEMS BIOLOGY

Oligonucleotide microarrays. The most widely used methods to monitor the expression levels of RNA transcripts in a biological sample are based on microarrays. They measure the hybridization of fluorescently labeled cDNA, synthesized from extracted mRNA, to known nucleotide sequences spotted on solid surfaces[117]. For all genes on the microarray, an expression value is derived from the fluorescence intensity of the hybridized RNAs. These expression values are typically unitless and have meaning only in the context of a reference measurement. Before further analysis takes place, the measurements must therefore be normalized to remove systematic biases and to make it possible to compare measurements from different samples.

Quality assessment is likewise essential for the validity of later analyses. This is typically performed with the help of (platform-dependent) quality scores at the level of both individual probes and entire arrays, complemented by diagnostic visualization tools that have been developed for this purpose[118,119]. Evaluating the quality of individual arrays is routinely done with spatial intensity distributions plots and plots of intensity ratio versus mean intensity (**Supplementary Fig. 1a**). Comparison of multiple arrays can be achieved with intensity box plots, which are a practical tool to detect outlier arrays that should be excluded from subsequent analysis (**Supplementary Fig. 1b**). Several tools that provide quality assessment visualizations are listed in **Supplementary Table 1**.

RNA deep sequencing. The most recent transcriptomics approaches are based on the deep sequencing of transcripts extracted from biological samples[33]. The resulting sequence reads—typically 30 to 400 base pairs long, depending on the DNA-sequencing technology used—are then commonly aligned to a reference genome and evaluated to determine their quality.

Tools for data processing and quality assessment typically provide diagnostic visualizations. Examples include the R/Bioconductor packages ShortRead[120] and edgeR[121]. The latter provides many functions that are analogous to those in the limma package[122] for transcriptomics data from microarrays. Reads aligned to a genome can also be visualized and evaluated with some of the more recent genome browsers that can handle short read data, such as the Integrative Genomics Viewer (http://www.broadinstitute.org/igv/). This and similar tools are discussed in the accompanying review by Nielsen *et al.*[72].

Mass spectrometry. In mass spectrometry (MS) experiments, the compounds present in a sample are identified through the accurate measurements of their mass-to-charge ratios. MS has applications in many fields, including proteomics, metabolomics and interactome mapping.

In proteomic applications, typical MS data sets consist of lists of proteolytic peptides characterized by their mass-to-charge ratios (MS spectra, MS1). These peptides can be further fragmented and measurements of the resulting mass spectra (MS-MS spectra or tandem MS spectra, MS2) used to deduce their sequences. In some cases, complex samples must be fractionated and proteolytic peptides are separated using high performance liquid chromatography (LC) before MS analysis (LC-MS).

Several search engines have been developed to predict peptides and proteins through the comparison of experimentally measured spectra to theoretical spectra (predicted from sequence databases). Quality scores provide a measure of the reliability of a given protein or peptide identification[123]. For example, for Mascot[124], the most broadly used algorithm, the score features the number of identified peptides (sequence coverage).

The overall quality of entire MS data sets is generally measured by the false discovery rate (FDR), which is the 'expected' proportion of incorrect assignments among the accepted assignments. The most popular approach to calculate FDR is based on the use of a target-decoy database[123]. Also, an array of visualization tools has been developed to evaluate the technical quality of the samples and of MS runs. For example, the overall distribution of peptides in an LC-MS map can be visualized with Pep3D[125] or TOPPView[126], enabling the detection of possible biases, the presence of chemical contaminants or poor separations during the LC (**Supplementary Fig. 2**). Additionally, Pep3D can integrate quality scores for individual protein or peptide identifications generated by search engines into these maps (**Supplementary Fig. 3**). We list tools for mass spectrometry data visualization and evaluation in **Supplementary Table 1**.

In metabolomics applications, owing to large chemical diversity and variation in molecular composition of the analytes, various chromatographic systems, such as gas chromatography (GC), LC or electrochemistry (EC), are generally applied before MS. GC-MS is the most popular method for global metabolite profiling[127]. It can be complemented with LC-MS analysis to identify compounds that are not suitable for GC-MS analysis[128]. Similarly to the approaches developed for peptides, metabolites can be identified on the basis of their fragmentation patterns, for which mass spectral fingerprint libraries are being developed. Because the raw data are of the same kind as in proteomics mass spectrometry studies, very similar visualization methods are used to assess data quality (**Supplementary Table 1**).

Nuclear magnetic resonance. Nuclear magnetic resonance (NMR) is a common method in metabolomics and, in contrast to MS-based approaches, in most cases does not require analyte separation. NMR spectroscopy can provide detailed information on the molecular structure of compounds found in complex mixtures, and a wide range of small molecule metabolites in a sample can be detected simultaneously. Biofluids, cell and tissue extracts can be analyzed with minimal sample preparation through the use of ^{1}H NMR spectroscopy[129]. With the use of two-dimensional NMR spectra, the identification and reliable quantification of individual metabolites becomes feasible, which enables NMR-based metabolite profiling. Data processing and spectral deconvolution are challenging, and databases of NMR spectra of pure metabolites are not yet comprehensive, but they nonetheless do already help in the identification process[130]. Applications such as MetaboMiner[131] can be used for the semiautomated identification of metabolites in two-dimensional NMR spectra, supported by visualizations that allow the scientist to inspect the matches of peaks to reference spectra and assess match quality.

perturbations (for example, gene deletions, gene insertions and siRNA knockdowns). Often, these experiments produce new knowledge that is then either added to existing pathways or used to create new pathways. Thus, we end with a discussion of methods and tools for pathway editing.

Protein interaction data

A range of experimental methods are at present being used for high-throughput studies of protein interactions[8]. For instance, in yeast, pairwise interactions have been studied on the genome scale using yeast two-hybrid screens or protein complementation assays, whereas the assembly of proteins within complexes has been systematically charted using tandem affinity purification coupled with mass spectrometry (TAP-MS) (**Box 1** and **Supplementary Figs. 2** and **3**). Recent analyses have estimated that, in yeast, some 20,000 pairwise interactions may take place between the ~5,000 gene products[9], and about 800 protein complexes may exist[10]. As a result of these and similar studies in other species, vast amounts of protein interaction data are accumulating in public databases[11,12] such as DIP[13], HPRD[14] and IntAct[15].

The size and complexity of these data sets can be daunting; hence, a common general strategy is to iteratively dissect the data sets into smaller subsets. Typically, these subsets are defined as sets of proteins that belong to the same complex, or that are found at the same subcellular location, or that belong to a similar functional category. Visualization is key in this strategy, as human judgment and intervention are often needed, in part because of errors (false positives and false negatives) in protein interaction data sets[9,16]. Many visualization tools have been developed specifically to support the analysis of protein networks (**Table 1**); here we discuss how these tools can be used to help dissect large data sets of interactions, extract biological insight and generate hypotheses leading to further experimental investigations.

As proteins rarely act alone, a first step in analyzing a protein interaction data set is to identify protein complexes and groups of complexes. For small, simple networks, visualized as a graph in which each node represents a protein and each edge represents an interaction between two proteins, the arrangement of proteins and complexes can usually be seen clearly using a standard 'force-directed' layout[17], which automatically arranges each node

Table 1 | Visualization tools focused on interaction networks

Name	Cost	OS	Description	URL
Stand-alone				
Arena 3D[63]	Free	Win, Mac, Linux	Visualization of biological multi-layer networks in 3D	http://www.arena3d.org/
BiNA[81]	Free	Win, Mac, Linux	Exploration and interactive visualization of pathways	http://www.bnplusplus.org/bina/
BioLayout Express 3D[37]	Free	Win, Mac, Linux	Generation and cluster analysis of networks with 2D/3D visualization	http://www.biolayout.org/
BiologicalNetworks[82]	Free	Win, Mac, Linux	Analysis suite; visualizes networks and heat map; abundance data	http://www.biologicalnetworks.org/
Cytoscape*[20,83]	Free	Win, Mac, Linux	Network analysis; extensive list of plug-ins for advanced visualization	http://www.cytoscape.org/
GENeVis[36]	Free	Win, Mac, Linux	Network and pathway visualization; abundance data	http://tinyurl.com/genevis/
Medusa[84]	Free	Win, Mac, Linux	Basic network visualization tool	http://coot.embl.de/medusa/
N-Browse[85]	Free	Win, Mac, Linux	Network visualization software for heterogeneous interaction data	http://www.gnetbrowse.org/
NAViGaTOR[23,86]	Free	Win, Mac, Linux	Visualization of large protein-interaction data sets; abundance data	http://tinyurl.com/navigator1/
Ondex[87]	Free	Win, Mac, Linux	Integrative workbench: large network visualizations; abundance data	http://www.ondex.org/
Osprey[88]	Free	Win, Mac, Linux	Tool for visualization of interaction networks	http://tinyurl.com/osprey1/
Pajek[89]	Free	Win	Generic network visualization and analysis tool	http://pajek.imfm.si/
ProViz	Free	Win, Mac, Linux	Software for visualization and exploration of interaction networks	http://tinyurl.com/proviz/
SpectralNET[90]	Free	Win	Network visualizations; scatter plots for dimensionality reduction methods	http://tinyurl.com/spectralnet/
Tulip[91]	Free	Win, Mac, Linux	Generic visualization tool; extremely large networks; 3D support	http://tulip.labri.fr/TulipDrupal/
VANTED[21]	Free	Win, Mac, Linux	Combined visualization of abundance data, networks and pathways	http://tinyurl.com/vanted/
yEd	Free	Win, Mac, Linux	Generic network visualization software; offers many layout algorithms	http://tinyurl.com/yEdGraph/
Cytoscape plug-in				
BiNoM[92]	Free	Win, Mac, Linux	Extensive support for common systems biology network formats	http://tinyurl.com/binom1/
BioModules[24]	Free	Win, Mac, Linux	Detects modules in networks; maps abundance data onto nodes and modules	http://tinyurl.com/biomodules/
Cerebral*[26,78]	Free	Win, Mac, Linux	Biologically motivated layout algorithm; maps abundance data; clustering	http://tinyurl.com/cerebral1/
MCODE[18]	Free	Win, Mac, Linux	Network clustering algorithm; support for manual cluster refinement	http://tinyurl.com/MCODE123/
VistaClara[42]	Free	Win, Mac, Linux	Mapping of abundance data to nodes and 'heat strips'; provides heat map	http://tinyurl.com/cytoplugins/
Web-based				
Graphle[93]	Free		Distributed client/server network exploration and visualization tool	http://tinyurl.com/graphle/
Lichen	Free		Library for web-based visualization of network and abundance matrix data	http://tinyurl.com/Lichen1/
MAGGIE Data Viewer	Free		Visualization of networks; abundance data in heat maps and profile plots	http://maggie.systemsbiology.net/
STITCH[31]	Free		Construction and visualization of networks from a wide range of sources	http://stitch.embl.de/
VisANT[22]	Free	Win, Mac, Linux	Analysis, mining and visualization of pathways and integrated omics data	http://visant.bu.edu/

Some of the tools in this table have capabilities similar to tools that are listed in other tables. To avoid listing tools in more than one table, we assigned tools to tables on the basis of what we understand to be their primary purpose. *Our recommendations. Free means the tool is free for academic use; $ means there is a cost. OS, operating system: Win, Microsoft Windows; Mac, Macintosh OS X. Tools running on Linux usually also run on other versions of Unix. 2D, two-dimensional; 3D, three-dimensional.

to minimize the number of edge crossings while trying to keep the lengths of all edges approximately the same. However, this approach quickly becomes inadequate as the network size and complexity increase (**Fig. 1a**). Instead, clustering approaches are used, which predict higher-order protein complexes from the interaction data. One very commonly used tool for this purpose is MCODE[18]. For TAP-MS and other data sets where components of protein complexes are experimentally determined, other clustering methods are used (for example, 'clique percolation'[19]). The results of these clustering analyses can then be used to change the layout and appearance of the network (**Fig. 1a,b**) in a way that may yield biological insights that cannot be easily obtained by simply examining lists of proteins or protein complexes. For instance, by viewing the network, the scientist may notice connections between two complexes that suggest a previously unknown biological relationship. Furthermore, on the basis of previous knowledge, the scientist may be able to assign a putative function or subcellular localization to the complex; this information can be visualized using node color or shape to represent the functional category or location of the proteins. Similarly, node color or shape can be used to show which proteins belong to the same complex (**Fig. 1b**).

Most network visualization tools provide the ability to interactively change the layout of the network—for example, by automatically arranging a user-defined group of proteins into any of a variety of arrangements (a circle, a line and so forth) or by manually moving nodes. This ability can be very useful in creating visualizations that emphasize biologically significant relationships and interactions between complexes (**Fig. 1c**) or between 'hub' proteins and their partners (for example, between kinases and their substrates). Tools that support such interactive editing particularly well include Cytoscape[20], VANTED[21], VisANT[22] and NAViGaTOR[23].

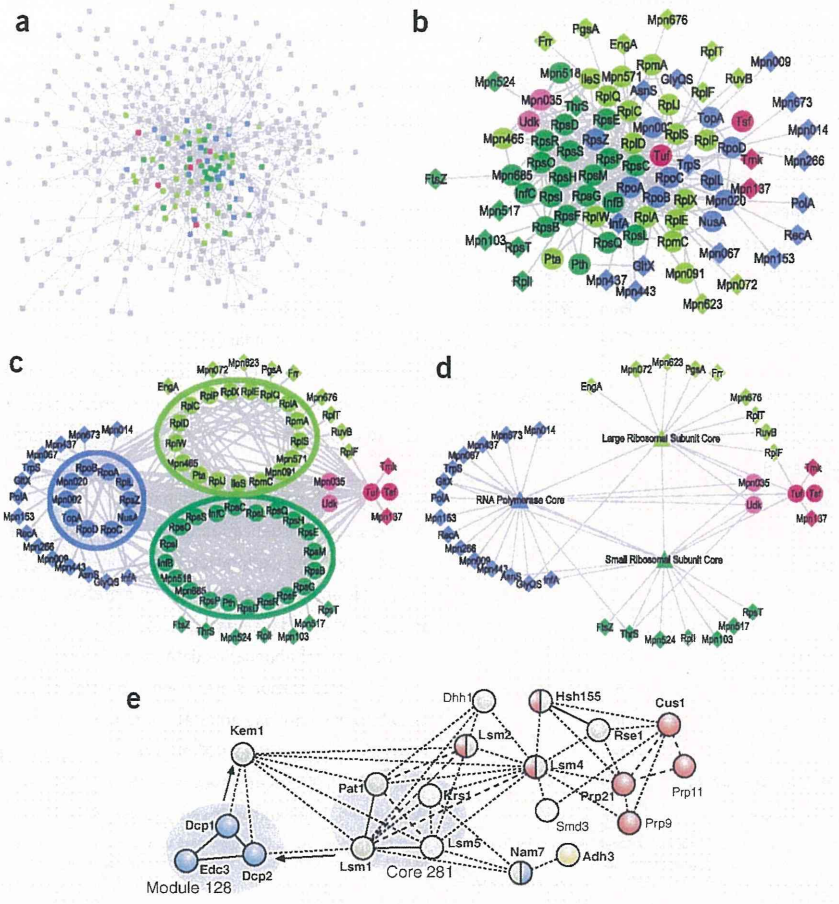


Figure 1 | Visualization of protein interaction networks. (**a**–**d**) Cytoscape[20] images of *Mycoplasma pneumoniae* protein interaction data derived by mass spectrometry[19] analysis. (**a**) Initial protein interaction network (>400 proteins) laid out with a force-directed algorithm. Nodes discussed in the following steps are overlaid with functional annotations (blue, RNA polymerase; dark or light green, small or large ribosomal subunits, respectively; red, elongation factor). (**b**) Recomputed network remaining after removal of nodes not of interest. Five computationally determined complexes are colored according to functional annotation. Node shapes represent different roles in the complex (circle, core protein of complex; diamond, protein attached to complex but not part of the core). At this stage, clusters emerge. (**c**) Manual refinement of the network layout emphasizing structure of protein complexes and interactions between them. (**d**) Collapse of nodes in each complex core, simplifying the network and emphasizing global properties. (**e**) Stages in deadenylation-dependent mRNA degradation in *Saccharomyces cerevisiae*. Reproduced from Gavin *et al.*[10]. Arrows show the order of sequential steps in a cellular process. Proteins are colored according to their localization (green, cytoplasm; red, nucleus; blue, punctate composite (undefined subcellular structure); yellow, mitochondria; white, unknown). Edge styles represent socio-affinity indices (dotted, 5–10; dashed, 10–15; solid, >15). TAP-MS bait proteins, bold; shaded circles, protein complexes.

It is often useful to collapse all members of a protein complex or cluster into a single 'meta-node' (**Fig. 1d**) that can later be expanded, depending on screen space and the desired level of detail. Meta-nodes not only simplify the appearance of the network, they can also be useful in more clearly illustrating biological relationships between protein complexes. Meta-nodes can also help to visually arrange the network to give insight into the integration and coordination of cellular functions (**Fig. 1d**). Meta-nodes are supported by yEd (http://tinyurl.com/yEdGraph/), BioModules[24] and VisANT, the last of which further allows meta-nodes to be nested hierarchically and can show 'meta-edges'[25] between meta-nodes—these can indicate, for example, when proteins are shared between two collapsed complexes.

Present high-throughput experimental methods often do not determine the spatial, or subcellular, location where an interaction takes place, so it can be highly informative to include any previous protein localization information in the analysis of these data sets. For instance, the network may be filtered to show only proteins known to occur in selected locations, thus simplifying it and allowing the scientist to focus only on interactions within a defined subcellular location. Alternatively, subcellular location

Table 2 | Visualization tools focused on pathways

Name	Cost	OS	Description	URL
Stand-alone				
BioTapestry[94]	Free	Win, Mac, Linux	Visualization of genetic regulatory networks, also with experimental data	http://www.biotapestry.org/
Caleydo[95]	Free	Win, Linux	Interactive framework for pathway and expression data; 3D 'bucket' view	http://www.caleydo.org/
CellDesigner*[51]	Free	Win, Mac, Linux	Drawing and simulation of pathways and models; supports SBGN	http://www.celldesigner.org/
Edinburgh Pathway Editor	Free	Win, Mac, Linux	Construction and visualization of pathway diagrams; supports SBGN	http://tinyurl.com/EdinburghPE/
GenMAPP[40]	Free	Win	Pathway visualization and construction; abundance data	http://www.genmapp.org/
IngenuityPathways	$	Win, Mac, Linux	Full analysis suite; network and pathway visualizations; abundance data	http://tinyurl.com/IngenuityPath/
JDesigner[52]	Free	Win	Drawing and simulation of pathways and models	http://tinyurl.com/jdesigner/
KaPPA View[48]	Free	Win	Analysis and visualization of plant pathways and mapped abundance data	http://tinyurl.com/kappa-view/
KEGG Atlas[96]	Free	Win, Mac, Linux	Visualization of abundance data on interactive KEGG pathways	http://www.genome.jp/kegg/
MetaCore	$	Win, Mac, Linux	Pathway, network and omics data analysis and visualization suite	http://www.genego.com/
PathVisio[97]	Free	Win, Mac, Linux	Pathway visualization and editing; supports mapping of omics data	http://www.pathvisio.org/
VitaPad[98]	Free	Win, Mac, Linux	Editing of pathway diagrams; integration of abundance data	http://tinyurl.com/vitapad/
Web-based				
ArrayXPath[99]	Free		Mapping of abundance data to pathway visualizations	http://tinyurl.com/ArrayXPath/
GEPAT[100]	Free		Analysis suite; visualization of transcriptomics data on pathways maps	http://tinyurl.com/GEPAT1/
iPath[101]	Free		Visualization and exploration of combined KEGG pathways	http://pathways.embl.de/
MapMan[46]	Free		Visualization of abundance data on metabolic pathways	http://tinyurl.com/MapManApp/
Omics Viewer[47,102]	Free		Mapping of abundance data to BioCyc pathway diagrams	http://www.biocyc.org/
Pathway Explorer[49]	Free		Visualization of abundance data on pathways	http://tinyurl.com/pathwayexp/
PATIKA[103]	Free		Pathway visualization suite; good support for signaling pathways	http://www.patika.org/
Payaologue	Free		Collaborative pathway annotation and visualization tool	http://celldesigner.org/payao/
ProMeTra[41]	Free		Maps abundance matrices of multiple data types to pathways	http://tinyurl.com/ProMeTra/
Reactome SkyPainter[30]	Free		Visualization of over-represented pathways and reactions from gene lists	http://reactome.org/
WikiPathways[62]	Free		Wiki-based, community-driven pathway curation and visualization tool	http://www.wikipathways.org/

Some of the tools in this table have capabilities similar to tools that are listed in other tables. To avoid listing tools in more than one table, we assigned tools to tables on the basis of what we understand to be their primary purpose. *Our recommendations. Free means the tool is free for academic use; $ means there is a cost. OS, operating system: Win, Microsoft Windows; Mac, Macintosh OS X. Tools running on Linux usually also run on other versions of Unix. 3D, three-dimensional.

can be indicated using node coloring; this can be particularly useful when studying the interactions of complexes that move between subcellular locations (**Fig. 1e**). Another common strategy is to arrange the network so that all proteins belonging to the same subcellular location are gathered together in one region (**Fig. 2, see 2a**). For small networks, such a layout depicting subcellular localization is often created manually. However, for large networks, it is much more convenient to use tools that can achieve such a layout automatically, such as Cerebral[26] (**Fig. 2a**) and PATIKA[27]. These tools also draw boundaries or use shading so that the scientist can see clearly which regions of the network correspond to which subcellular locations.

Protein interaction data sets commonly do not capture information about dynamic changes in protein abundance. Thus, as with spatial information, it is often useful to include temporal information from other experiments—for example, by identifying proteins whose abundance is known to vary throughout the cell cycle. This information can be used to simplify a large network by either depicting only proteins that are coexpressed or by mapping expression or abundance profiles of proteins of interest onto nodes[28] in the network, as described in more detail below (see Network enrichment).

These processes of dissection are all aimed at dividing a protein interaction data set into manageable, biologically significant parts that can be interpreted; during this process of interpretation, a scientist often makes use of previously established knowledge, particularly pathways (for example, KEGG[29] or Reactome[30]) and networks (for example, STITCH[31]). In some cases, to illustrate a result or insight, it can be useful to add interactions derived from previous studies—thus forming a hybrid network that shows both new and old data (**Fig. 1e**).

Expression profile data

A range of experimental methods are being used for high-throughput expression profiling (**Box 1**, **Supplementary Fig. 1** and **Supplementary Table 1**); in addition to gene expression profiling with DNA microarrays[32] and RNA deep sequencing[33], a promising emerging technology is quantitative protein expression profiling based on mass spectrometry[34,35]. Gene expression profile data sets are being deposited in two main repositories, ArrayExpress (http://www.ebi.ac.uk/arrayexpress/) and Gene Expression Omnibus (http://www.ncbi.nlm.nih.gov/geo/), with around 15,000 studies now in the public domain.

The initial goal in analyzing expression profiles is usually to find a set of genes or, less typically, proteins that share a related pattern of expression—for example, genes that are up- or down-regulated in a certain genotype, disease model or human disease, or in response to a drug treatment. The challenge is that a single data set may contain expression profiles for over 10,000 genes, measured over a range of time points and experimental conditions, so that determining which genes are potentially relevant to the studied problem requires an extensive search through a large amount of often noisy, multivariate data. Together with various clustering algorithms, visualization is key in these analyses[32], and

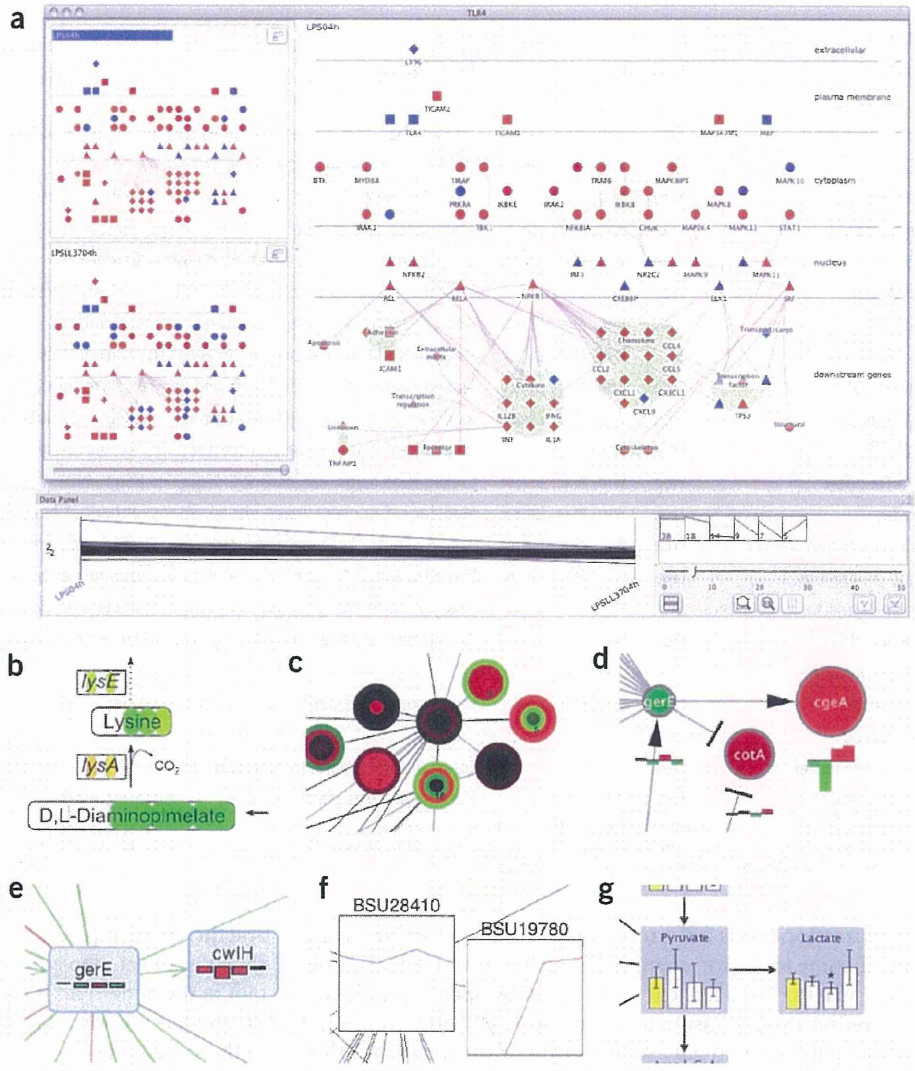


Figure 2 | Omics data overlaid onto biological networks. (**a**) Cerebral[78] showing the TLR4-to-NF-κB signaling pathway[79] laid out according to subcellular localization and functional annotation (green shading). Direction of information flow is from top to bottom. Node colors represent relative expression (red, upregulation; blue, downregulation) and edge colors represent interaction type (orange, phosphorylation; cyan, other protein interaction; purple, transcriptional regulation). The left two panels are a 'small multiples' display of the same pathway overlaid with gene expression data for two experimental conditions. In this case, the top panel has been selected, and hence is also shown in the main window. The bottom panels show the detailed expression profiles corresponding to genes shown in the pathway panel (see 'Network enrichment'). The data set shows how upregulation of NFKB1 explains the observed upregulation of several chemokine proteins. (**b**) ProMeTra[41] display showing both metabolomics and transcriptomics time series data from five time points. Metabolite and enzyme nodes in the pathway map are subdivided into five areas, one per time point. Areas are color coded (green, upregulation; yellow, no change; white, missing data) to indicate metabolite concentrations and transcript levels relative to a reference time point. (**c**) Lichen rendering of a gene regulatory network overlaid with transcriptomics data using a circular heat map. Each concentric ring represents a time point, and the color of the circle represents expression level (red, upregulation; green, downregulation; black, no change). Numbers identify genes. (**d**) VistaClara[42] display showing transcript levels across four time points relative to a reference time point as 'heat strips' below the nodes (height of bar, relative expression; red, upregulation; green, downregulation). Node color indicates expression level from time point 4. Node size indicates reliability of the measurement taken at time point 4. (**e**) GENeVis[36] visualization of the same data set. Color coding as in **d**; height of bar corresponds directly to the reliability of the measurement at each time point. (**f**) Profile plots of the same data embedded in nodes in VisANT[22]. The color of each line segment represents the change in expression levels between two time points (red and purple, increase; blue and cyan, decrease). (**g**) Visualization of metabolite concentrations in a pathway map in VANTED[21] using a bar chart with error bars.

a wide range of tools have been developed to aid the visualization process (**Table 3**). Many of these tools implement a set of commonly applied methods (**Box 2** and **Fig. 3**); in particular, scatter plots combined with dimensionality reduction (**Fig. 3a**), profile plots (**Fig. 3b**), heat maps, and dendrograms (**Fig. 3c**), as well as clustering. As microarray gene expression analysis has matured as an experimental technique, many of the corresponding visualization methods have become well established and are widely used.

Network enrichment. Once a list of potentially relevant genes has been found using the above types of analysis, the next task is often to find pathways or networks where these genes are significantly over-represented. These 'enrichment' searches can be launched directly from several network visualization tools, for example, GENeVis[36], Reactome SkyPainter[30], Metacore (GeneGo Inc.) or BioLayout Express 3D[37]. A logical next step is then to map gene expression levels onto the identified pathways. Interpreting expression data in the context of a visualized pathway or network usually proves more insightful than without this type of information. For instance, visualizing the data in the context of pathways may show how the upregulation of a transcription factor explains the upregulation of many other genes under its control (**Fig. 2a**) and may lead to testable experimental hypotheses.

A wide range of representations are used for mapping gene expression levels onto pathways and networks, with the ideal choice depending on the specific experiment and question of interest[38] (**Fig. 2**). A simple approach that is available as part of many tools (**Table 1**) is to represent expression levels as a color gradient, as in a heat map, and then color the nodes in the network according to their expression level under a particular condition (**Fig. 2a**).

Table 3 | Visualization tools for multivariate omics data

Name	Cost	OS	Description	URL
Stand-alone				
BicOverlapper[104]	Free	Win, Mac, Linux	Visualization of biclusters combined with profile plots and heat maps	http://vis.usal.es/bicoverlapper/
BiGGEsTS[105]	Free	Win, Mac, Linux	Heat map–based bicluster visualization	http://tinyurl.com/BiGGEsTS/
Brain Explorer[76]	Free	Win, Mac	Visualization of 3D transcription data in the central nervous system	http://tinyurl.com/brainExplorer/
Caryoscope[75]	Free	Win, Mac, Linux	Abundance data mapped to chromosomal location	http://tinyurl.com/caryoscope/
Data Matrix Viewer	Free	Win, Mac, Linux	Simple profile plot visualization; supports Gaggle	http://gaggle.systemsbiology.net/
EXPANDER[106]	Free	Win, Linux	Heat maps, scatter plots and profile plots of cluster averages	http://acgt.cs.tau.ac.il/expander/
Genesis[107]	Free	Win, Mac, Linux	Analysis suite; offers several interactive visualizations	http://tinyurl.com/genesisclient/
GeneSpring GX*	$	Win, Mac, Linux	Analysis suite; interactive and linked visualizations; also networks	http://tinyurl.com/genespring/
GeneVAnD[108]	Free	Win, Mac, Linux	Linked heat maps, dendrograms and 2D/3D scatter plots	http://tinyurl.com/GeneVAnD/
geWorkbench	Free	Win, Mac, Linux	Modular suite; heat maps, dendrograms, profile and scatter plots	http://tinyurl.com/geWorkbench/
HCE*[109]	Free	Win	Linked heat map, profile and scatter plots; systematic exploration	http://tinyurl.com/HCExplorer/
Java TreeView*[110]	Free	Win, Mac, Linux	Linked heat maps, karyoscopes, sequence alignments, scatter plots	http://jtreeview.sourceforge.net/
Mayday[111]	Free	Win, Mac, Linux	Modular suite; many linked visualizations; enhanced heat map[112]	http://tinyurl.com/maydaywp/
MultiExperiment Viewer*[113]	Free	Win, Mac, Linux	Analysis suite; heat maps, dendrograms, profile and scatter plots	http://www.tm4.org/
PointCloudXplore[77]	Free	Win, Mac, Linux	Visualization of 3D transcription data in *Drosophila* embryos	http://tinyurl.com/PointCloudXplore/
Spotfire Functional Genomics	$	Win	Analysis suite; many linked visualizations and exploration tools	http://spotfire.tibco.com/
TimeSearcher[114]	Free	Win	Exploration and analysis of time series; advanced profile plots	http://tinyurl.com/timesearcher/
R/BioConductor				
Geneplotter	Free	Win, Mac, Linux	Karyoscope-style plots and other visualizations	http://www.bioconductor.org/
Web-based				
ExpressionProfiler[115]	Free		Transcriptomics data analysis suite with basic visualizations	http://tinyurl.com/expresspro/
GenePattern[116]	Free		Modular analysis platform; several visualization modules available	http://tinyurl.com/GenePatt/

Some of the tools in this table have capabilities similar to tools that are listed in other tables. To avoid listing tools in more than one table, we assigned tools to tables on the basis of what we understand to be their primary purpose. *Our recommendations. Free means the tool is free for academic use; $ means there is a cost. OS, operating system: Win, Microsoft Windows; Mac, Macintosh OS X. Tools running on Linux usually also run on other versions of Unix. 2D, two-dimensional; 3D, three-dimensional.

If expression levels from more than one condition are being studied, some tools (for example, VisANT and VistaClara) allow the scientist to visualize them sequentially, by updating node colors to reflect the expression levels of a selected condition. Some tools switch automatically to depiction of the next condition after a predefined time interval, which leads to an animation-like visualization that is well suited to interpreting data from a time series. An alternative strategy to viewing the data from different conditions in series is to view them in parallel, by arranging multiple versions of the same network in a grid, where each version represents the expression levels (visualized as node color) for one condition or time point. This approach is known as 'small multiples'[39] and allows the scientist to visually compare expression levels between conditions, which is not well supported by animation. A well-designed implementation of small multiples is available in Cerebral[26] (**Fig. 2a**).

Besides animation and small multiples, a third approach is to show the complete expression profile within the nodes of a network. The most common representation of this type is based on a miniature heat map embedded in each node (**Fig. 2b**) and is available in several tools, including GenMAPP 2 (ref. 40), GeneSpring GX (Agilent Technologies) and ProMeTra[41]. The Lichen package (http://tinyurl.com/Lichen1/) uses a circular heat map to depict this information, which has the advantage of being very compact (**Fig. 2c**). VistaClara[42] provides 'heat strips', in which the heights of the bars as well as their colors correspond to the expression levels (**Fig. 2d**). In contrast, in GENeVis, bar heights represent confidence measures, so that reliable measurements are taller and therefore emphasized (**Fig. 2e**). A less common alternative to showing a heat map embedded in the node is to embed a profile plot in the node. This is supported by, for instance, VisANT (**Fig. 2f**) and has the advantage that multiple profiles can be displayed in the same node—for example, when a meta-node represents a set of genes. In VANTED, each node of the network has embedded visualizations of exceptionally high detail, showing legends, grid lines, bar charts or error bars (**Fig. 2g**). Although powerful, this representation requires the nodes to be rather large in order to show the details of the embedded visualizations, which effectively limits its application to only small pathways and networks.

For expression profiles with many conditions, visualizing all these data directly in the network is invariably problematic because of a lack of space, and an approach that links visualization of the network to a separate visualization of the expression profiles is more appropriate. In the linked approach, a heat map (as implemented in VistaClara) or a profile plot (as implemented in Cerebral; **Fig. 2a**) is shown next to the network and when the scientist selects nodes in the network, the corresponding expression profiles are highlighted in the linked heat map or profile plot, or vice versa. This approach allows the scientist to check, for instance, whether the members of a putative protein complex in the network visualization are coexpressed, by comparing the corresponding gene expression profiles in the linked heat map. Conversely, selection of a set of coexpressed genes in a clustered heat map would allow exploration of their role in the linked protein interaction network: the scientist could directly see whether these genes are part of the same complex, what their interaction

BOX 2 KEY VISUALIZATION METHODS FOR MULTIVARIATE DATA

Multivariate data, for instance from gene expression studies, are very common in systems biology, and many tools have been developed to analyze and visualize such data (**Table 3**). The three most commonly used visualization methods are scatter plots, profile plots and heat maps.

Scatter plots. Scatter plots (**Fig. 3a**) are primarily used to examine dependencies between two variables, but in combination with dimensionality reduction methods, they can also be applied to multivariate data. For instance, to gain insight into the global patterns in a gene expression matrix, a dimensionality reduction method may be applied to obtain a two-dimensional (sometimes three-dimensional) representation of the expression profiles, which are then visualized in a scatter plot to reveal clusters and outliers in the data. Some frequently applied dimensionality reduction methods for this purpose are principal component analysis[132] (PCA) and multi-dimensional scaling[133] (MDS), which are implemented in many tools. Besides PCA and MDS, many other suitable dimensionality reduction methods exist[134], but they are often not easily accessible to the casual user.

Scatter plots combined with dimensionality reduction methods are an excellent tool for gaining insight into the overall structure of large sets of expression profiles. However, because of the dimensionality reduction itself, it is not possible to extract information about the relationship between expression levels and the conditions under study.

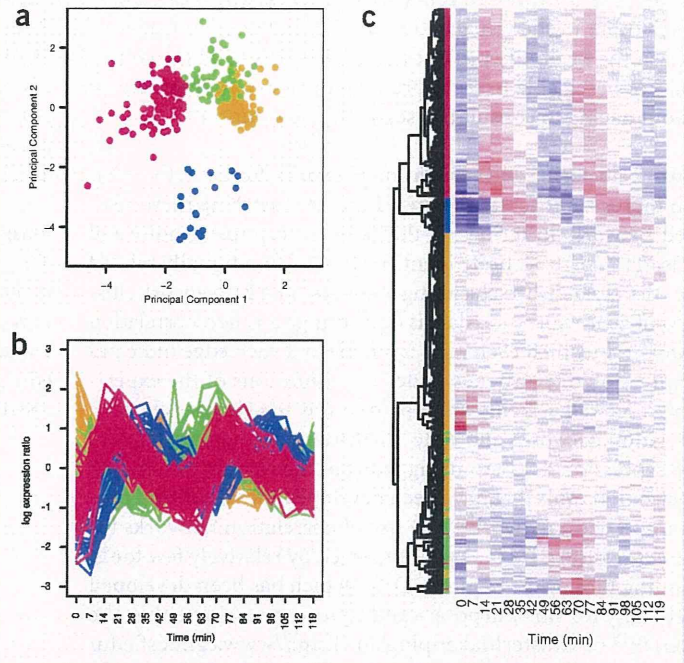


Figure 3 | Visualization of gene expression profiles. Expression of 320 transcripts from *S. cerevisiae*, collected over 18 time points throughout the cell cycle[80]. Colors indicate cluster membership based on a *k*-means clustering ($k = 4$). (**a**) Scatter plot showing a projection of the profiles on the first two principal components obtained by PCA. (**b**) Profile plot of gene expression across all 18 time points, including *k*-means cluster information. Genes in the red and blue clusters appear active in the G1 and S phase of the cell cycle, respectively. Phase assignments for yellow and green clusters are unclear. (**c**) Heat map of the profiles. Colors represent abundance (red, higher than control; blue, lower than control; white, no change). Rows of the heat map have been reordered according to a hierarchical clustering, represented by the dendrogram. The color bars between the dendrogram and heat map indicate the *k*-means clusters, allowing comparison of the two clustering results. Images made with R (http://www.r-project.org/).

Profile plots. Profile plots (**Fig. 3b**), also known as parallel coordinate plots[135], visualize the expression levels of a large number of transcripts across all samples. Thus, they provide insight into the patterns of correlation between samples and expression levels. For instance, at a glance, the scientist can determine whether a transcript is expressed constitutively in all conditions or whether it is only expressed in a single condition, such as a particular tissue or phase of the cell cycle. Furthermore, it is possible to generate hypotheses about trends, such as increasing expression levels for a transcript over time after a stimulus, or differential expression of a transcript—for instance, between samples of diseased and normal tissue. Because many profiles are shown in the same plot, the scientist can interpret such observations in the context of the overall data set.

A profile plot can also be queried visually for transcripts with a particular behavior, such as low expression in one set of samples and high in another set, or for profiles that are similar to that of a transcript of interest. A substantial disadvantage of profile plots is that, owing to the manner in which they are constructed, profiles overlap, severely limiting the number of profiles that can be visualized effectively at the same time.

Heat maps. Heat maps[136,137] (**Fig. 3c**) are the most commonly used visualization method for expression matrices[138] and can be generated using most tools. Like profile plots, heat maps visualize the abundance of each transcript in each sample, but the profiles do not overlap, which means that more profiles can be visualized effectively. However, the size of the heat map grows with the number of profiles, so that the available screen space is often a limiting factor.

A key aspect of heat map visualization is the reordering of the rows, which ensures that similar profiles are placed near each other. Typically this reordering is done using hierarchical clustering[137], and a dendrogram showing the hierarchy is usually arranged immediately adjacent to the heat map (**Fig. 3**). This combined view helps a scientist to see groups of genes that have a similar expression pattern. The dendrogram conveys which genes are clustered together, and also which genes are outliers with an unusual expression pattern. The heat map allows the scientist to see in more detail which features of the expression pattern are shared by gene clusters. For example, genes in a cluster may have a peak expression at about the same time in an experiment.

partners are, or whether they are located in the same subcellular compartment. In contrast, when expression profiles are shown only in the nodes of the network, this type of analysis is not possible because coexpressed genes are not necessarily located next to each other in the visualization. However, there is a trade-off between the flexibility provided by linked views and the convenience of being able to see expression profiles and interactions without having to consult two separate visualizations.

Network clustering and correlation networks. Recently, there has been increased interest in a new kind of clustering method—called 'network clustering'[37]—that is less susceptible to noise and can lead to more accurate identification of functionally related genes than established clustering methods (**Box 2**). Network clustering of gene expression data is done using so-called 'correlation networks', in which each gene is a node and each edge indicates coexpression of two genes under the conditions of the experiment[43]. As well as being an improved way to calculate clusters, correlation networks allow the scientist to interactively explore gene expression data sets using many of the rich set of network visualization tools that have been developed for visualizing protein interaction networks. The use of correlation networks for gene expression data is as yet supported by relatively few tools, including BioLayout Express 3D[37]—which has been developed specifically for this purpose—and Cytoscape, using either the MCODE[18] or ClusterMaker plug-in (http://www.cgl.ucsf.edu/cytoscape/). However, we anticipate that correlation networks may become one of the established methods for interpreting gene expression data sets.

Metabolic profile data

A wide variety of spectroscopic methods are being used for high-throughput studies of small-molecule metabolites[44], two of the most popular being mass spectrometry and nuclear magnetic resonance spectroscopy (**Box 1** and **Supplementary Table 1**). Typically, present methods identify hundreds of metabolites per experiment. Additionally, many as-yet-unidentified compounds can be reproducibly detected. These experimental data are collected in several public repositories, the largest of which is now SetupX[45], containing ~20,000 samples from more than 300 studies.

The general goal in analyzing metabolite profiles is to gain detailed insight into the molecular mechanisms of cellular metabolic pathways. The identification of molecules that may be used as reliable biomarkers of disease is also of great interest. Metabolite profiles are typically analyzed to find sets of metabolites with similar profiles or to measure the impact of genetic modifications, drugs and other biotic or abiotic factors on the metabolome of an organism.

As with gene and protein profiles, visualization is key in these analyses, and the same, or very similar, methods (**Box 2**) and tools (**Table 3**) are typically used. As for gene expression data, one of the key visualization methods in metabolomics involves the enrichment of metabolic pathways with visualizations of metabolite concentrations (**Fig. 4a**), and often the same visual representations as for gene expression data can be used (**Fig. 2b–g**). Visualizing such enriched metabolic pathways can be very useful in understanding the concerted changes of metabolite pools within the cell. In addition, enriched pathways can help to identify metabolites that should be present according to enzymatic reactions contained in the metabolic pathway but that have not been detected in the measurements. If such metabolites are identified, further experiments attempting to detect these molecules can be conducted. Many visualization tools (for example, MapMan[46], Pathway Tools Omics Viewer[47], KaPPA-view[48], PathwayExplorer[49] and ProMeTra[41]) have been developed to facilitate enriched views of metabolic pathways, usually with close integration of metabolic pathway databases. These tools overlay metabolite profiles primarily on static images of pathways obtained from sources such as KEGG[29] or MetaCyc[50]-based databases.

Pathway editing

The analysis of new experimental data sets, as outlined above, usually produces new insight into biological processes, which may be used to modify existing pathways or to create new pathways. A wide range of tools is available that support pathway editing (**Table 2**); the choice of which tool to use depends on the specific requirements of the task at hand.

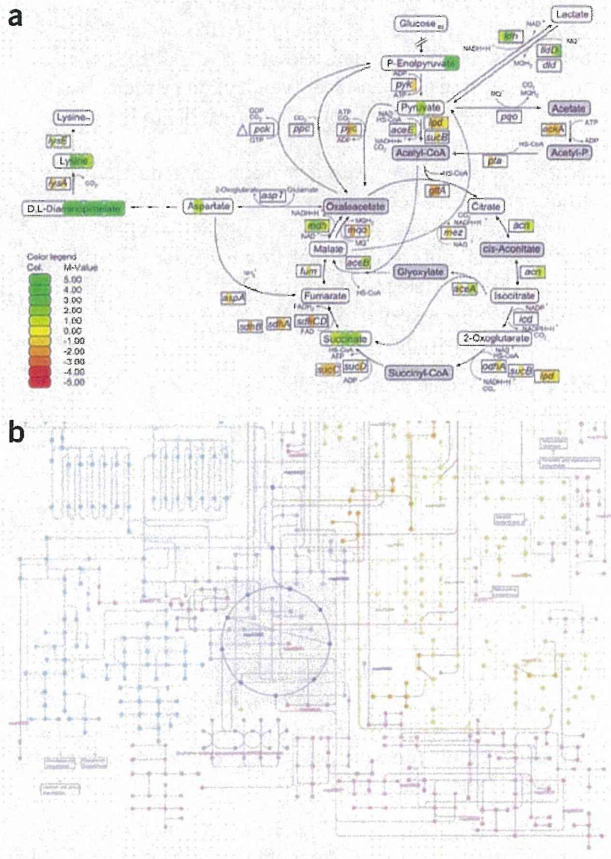


Figure 4 | Visualization of metabolic pathways and profile data. (**a**) A part of the glycolysis and citric acid cycle pathway in *Corynebacterium glutamicum* DM1730 overlaid with changes in metabolite concentrations and gene expression across five time points in relation to a reference time point. The visualization was created in ProMeTra[41] (as in **Fig. 2b**). Nodes shaded gray indicate metabolites for which no concentration data were available. (**b**) Enlarged iPath/KEGG Atlas image showing the glycolysis pathway in the context of other parts of the metabolic system. Yellow, amino acid metabolism, purple, energy metabolism. The shaded area corresponds to the citric acid cycle shown in **a**.

For building pathways from scratch or editing existing pathways, tools such as GenMAPP 2, PathVisio, and VANTED are useful, as they are designed for to assist the manual task of arranging nodes and edges. To this category also belong Cell Designer[51] and JDesigner[52], which further support pathway simulations by means of kinetic modeling. The insights gained from these simulations often lead to new hypotheses, which can then be tested in further experiments.

Manual layout of pathways quickly becomes tedious as the size of the pathway grows. Fortunately, a range of automated layout methods have been developed, each addressing specific layout requirements. Typically, these methods will arrange the network to highlight the overall state changes that occur—for example, making sure that all interactions point from left to right, and thus creating an overall causal flow from left to right. Automated layout can be particularly useful for updating large networks when new knowledge (nodes or interaction edges) becomes available. For example, PATIKA[53] has an automated layout method that shows the causal flow of events through different subcellular compartments. This is particularly useful for depicting signaling networks[27]. Although these specialized automated layout methods are useful, they are usually of low quality compared to manually laid out pathways created by human experts and often require manual editing in addition; however, judging by recent progress, we expect these method to continue to improve and to become increasingly useful[54].

For very large pathways, it can be important to use compact visual representations and pathway layouts that reduce the amount of detail shown. A very clear illustration of such a concise visual representation is iPath, which combines 120 KEGG pathways into a single, vast pathway map that provides an overview of all metabolism in an organism (**Fig. 4b**). Scientists can zoom into parts of the map to navigate to individual pathways.

Future perspectives

Systems biology is still rapidly evolving, which can make it difficult for tool developers to know which visualization tasks are the most important ones. However, as the field matures, the key tasks will likely become clearer, and the requirements and limitations of current visualization methods will become better understood[5]. This process will also be aided by insights from the emerging field of visual analytics[55], which specifically studies the role of visualization in the larger process of understanding and interpreting data. Visual analytics methods have begun to be applied to studying the connection between visualization and analytical reasoning in systems biology[5,56].

We anticipate that the near future will bring significant improvements in automated pathway and network layout to better match biologists' needs[54,57]. Innovation will continue to give more and better choices for the representations of nodes, edges and overlay information, as well as better ways to convey dynamic properties and to compare networks. Crucially, we expect that usability will improve, partly through improved navigation methods that help users manage large and complex networks[23,25,58].

Today, many tools for network and pathway visualization are stand-alone applications (**Tables 1** and **2**); however, there is a trend toward web-based applications, often coupled tightly to underlying databases. Web-based tools show great promise for facilitating collaboration between scientists at different locations[59–61], and several projects have recently been launched that are aimed at community-based collaborative editing of biological network data—notably Payaologue (http://celldesigner.org/payao/) and WikiPathways[62].

As experimental methods enable scientists to tackle larger and more complex systems, it is likely that significant innovations will be needed for visualizing future data sets. One possible direction for future network visualization tools would be to move beyond the standard two-dimensional layout, and tool developers are already exploring three-dimensional layouts (for example, BioLayout Express 3D[37]), combinations of both three-dimensional layouts and time (for example, E-Cell 3D, http://tinyurl.com/ecell3d/), or layouts that mix aspects of two and three dimensions (for example, Arena3D[63]). In addition, systems biologists may well be among the early adopters of innovations in hardware, such as multi-touch interfaces and larger, high-resolution displays[64].

As systems biology has evolved very quickly over the last decade, some of the difficulties faced by end-users today arise not from the intrinsic complexity of data but from a lack of standards. Biological pathways and networks are now distributed in over 300 web resources[4]—and in a field as interdisciplinary as systems biology, there is an obvious strength in such diversity. However, the field would clearly benefit from a parallel effort toward a consolidated resource, and we would like to add our voices to a call for a consolidated database, similar to the worldwide Protein Data Bank for three-dimensional structures[65].

The situation is somewhat better with file formats used to store interaction data, pathways and biochemical models. Although many formats are used, several have emerged as *de facto* standards for the exchange of pathway and network data—for example, PSI-MI[66] for protein interaction data, BioPAX (http://www.biopax.org/) for pathways and interaction networks, Systems Biology Markup Language (SBML)[67] for models of biochemical reactions and gene regulation and CellML[68] for exchange of a range of different biological models. In regard to graphical notation, there has recently been a significant community-driven proposal (Systems Biology Graphical Notation, SBGN[69]) toward developing a more unified standard, and several tools already support the creation and visualization of networks using this standard (see **Table 2**).

Ultimately, systems biology seeks to provide insights into the processes of organelles, cells, organs and even whole organisms. Fulfilling this ambitious goal requires still further development in visualization methods; in particular, better integration with visualization of other kinds of data, such as imaging data[70], macromolecular structures[71], genomes[72], and phylogenies[73]. Efforts to build such integrated visualization platforms have begun (for example, Visible Cell[74]), and in fact, many tools that bridge different data types and disciplines are already in place; for instance, there are tools that map transcript abundance (or, if available, protein abundance) onto chromosomal location[75] and onto three-dimensional anatomical representations of tissue[76,77]. However, truly integrated visualization of systems biology data across the entire range of possible data types is still very much in its infancy.

Note: Supplementary information is available on the Nature Methods website.

ACKNOWLEDGMENTS

The authors would like to acknowledge S. Kühner for providing the data for **Figure 1** and Â. Gonçalves for comments on parts of the manuscript. This work was partly supported by the European Union Framework Programme 6 grant 'TAMAHUD' (LSHC-CT-2007-037472).

COMPETING INTERESTS STATEMENT

The authors declare no competing financial interests.

Published online at http://www.nature.com/naturemethods/.
Reprints and permissions information is available online at http://npg.nature.com/reprintsandpermissions/.

1. Michal, G. *Biochemical Pathways: An Atlas of Biochemistry and Molecular Biology* (Wiley, New York, 1998).
2. Nishizuka, Y. The role of protein kinase C in cell surface signal transduction and tumour promotion. *Nature* **308**, 693–698 (1984).
3. Levine, M. & Davidson, E.H. Gene regulatory networks for development. *Proc. Natl. Acad. Sci. USA* **102**, 4936–4942 (2005).
4. Bader, G.D., Cary, M.P. & Sander, C. Pathguide: a pathway resource list. *Nucleic Acids Res.* **34** Database issue, D504–D506 (2006).
5. Saraiya, P., North, C. & Duca, K. Visualizing biological pathways: requirements analysis, systems evaluation, and research agenda. *Inf. Vis.* **4**, 191–205 (2005).
This paper represents one of the first attempts to critically evaluate the requirements for visualization software used in biology.
6. Suderman, M. & Hallett, M. Tools for visually exploring biological networks. *Bioinformatics* **23**, 2651–2659 (2007).
7. Pavlopoulos, G.A.G., Wegener, A.L.A. & Schneider, R.R. A survey of visualization tools for biological network analysis. *BioData Min.* **1**, 12 (2008).
8. Charbonnier, S., Gallego, O. & Gavin, A.C. The social network of a cell: recent advances in interactome mapping. *Biotechnol. Annu. Rev.* **14**, 1–28 (2008).
9. Yu, H. *et al.* High-quality binary protein interaction map of the yeast interactome network. *Science* **322**, 104–110 (2008).
10. Gavin, A.C. *et al.* Proteome survey reveals modularity of the yeast cell machinery. *Nature* **440**, 631–636 (2006).
11. Mathivanan, S. *et al.* An evaluation of human protein-protein interaction data in the public domain. *BMC Bioinformatics* **7** (suppl. 5), S19 (2006).
12. Ma'ayan, A. Network integration and graph analysis in mammalian molecular systems biology. *IET Syst. Biol.* **2**, 206–221 (2008).
13. Salwinski, L. *et al.* The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res.* **32** (database issue), D449–D451 (2004).
14. Prasad, T.S., Kandasamy, K. & Pandey, A. Human Protein Reference Database and Human Proteinpedia as discovery tools for systems biology. *Methods Mol. Biol.* **577**, 67–79 (2009).
15. Aranda, B. *et al.* The IntAct molecular interaction database in 2010. *Nucleic Acids Res.* **38** (database issue), D525–D531 (2010).
16. von Mering, C. *et al.* Comparative assessment of large-scale data sets of protein-protein interactions. *Nature* **417**, 399–403 (2002).
17. Fruchterman, T.M.J. & Reingold, E.M. Graph drawing by force-directed placement. *Software Pract. Exper.* **21**, 1129–1164 (1991).
18. Bader, G.D. & Hogue, C.W. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics* **4**, 2 (2003).
19. Kuhner, S. *et al.* Proteome organization in a genome-reduced bacterium. *Science* **326**, 1235–1240 (2009).
20. Shannon, P. *et al.* Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.* **13**, 2498–2504 (2003).
This paper describes the core Cytoscape software, which has become one of the most popular tools to visualize and analyze biological networks. This is partly due to the modular design of the software that allows developers to create plug-ins to address virtually any network analysis problem.
21. Junker, B.H., Klukas, C. & Schreiber, F. VANTED: a system for advanced data analysis and visualization in the context of biological networks. *BMC Bioinformatics* **7**, 109 (2006).
22. Hu, Z. *et al.* VisANT 3.5: multi-scale network visualization, analysis and inference based on the gene ontology. *Nucleic Acids Res.* **37** (web server issue), W115–W121 (2009).
23. McGuffin, M.J. & Jurisica, I. Interaction techniques for selecting and manipulating subgraphs in network visualizations. *IEEE Trans. Vis. Comput. Graph.* **15**, 937–944 (2009).
24. Prinz, S. *et al.* Control of yeast filamentous-form growth by modules in an integrated molecular network. *Genome Res.* **14**, 380–390 (2004).
25. Hu, Z. *et al.* Towards zoomable multidimensional maps of the cell. *Nat. Biotechnol.* **25**, 547–554 (2007).
26. Barsky, A., Munzner, T., Gardy, J. & Kincaid, R. Cerebral: visualizing multiple experimental conditions on a graph with biological context. *IEEE Trans. Vis. Comput. Graph.* **14**, 1253–1260 (2008).
27. Genc, B. and Dogrusoz, U. A layout algorithm for signaling pathways. *Inf. Sci.* **176**, 135–149 (2006)
28. de Lichtenberg, U., Jensen, L.J., Brunak, S. & Bork, P. Dynamic complex formation during the yeast cell cycle. *Science* **307**, 724–727 (2005).
29. Kanehisa, M. *et al.* KEGG for linking genomes to life and the environment. *Nucleic Acids Res.* **36** (database issue), D480–D484 (2008).
30. Matthews, L. *et al.* Reactome knowledgebase of human biological pathways and processes. *Nucleic Acids Res.* **37** (database issue), D619–D622 (2009).
31. Kuhn, M. *et al.* STITCH 2: an interaction network database for small molecules and proteins. *Nucleic Acids Res.* **38** (database issue), D552–D556 (2010).
32. Quackenbush, J. Computational analysis of microarray data. *Nat. Rev. Genet.* **2**, 418–427 (2001).
33. Wang, Z., Gerstein, M. & Snyder, M. RNA-Seq: a revolutionary tool for transcriptomics. *Nat. Rev. Genet.* **10**, 57–63 (2009).
34. Bantscheff, M., Schirle, M., Sweetman, G., Rick, J. & Kuster, B. Quantitative mass spectrometry in proteomics: a critical review. *Anal. Bioanal. Chem.* **389**, 1017–1031 (2007).
35. Gstaiger, M. & Aebersold, R. Applying mass spectrometry-based proteomics to genetics, genomics and network biology. *Nat. Rev. Genet.* **10**, 617–627 (2009).
36. Westenberg, M.A., van Hijum, S.A.F.T., Kuipers, O.P. & Roerdink, J.B.T.M. Visualizing genome expression and regulatory network dynamics in genomic and metabolic context. *Comput. Graph. Forum* **27**, 887–894 (2008).
37. Freeman, T.C. *et al.* Construction, visualisation, and clustering of transcription networks from microarray expression data. *PLOS Comput. Biol.* **3**, e206 (2007).
38. Saraiya, P., Lee, P. & North, C. in *IEEE Symp. Information Visualization* (InfoVis 2005) 225–232 (2005).
39. Tufte, E.R. *The Visual Display of Quantitative Information* 2nd edn. (Graphics Press, Cheshire, Connecticut, USA, 2001).
40. Salomonis, N. *et al.* GenMAPP 2: new features and resources for pathway analysis. *BMC Bioinformatics* **8**, 217 (2007).
41. Neuweger, H. *et al.* Visualizing post genomics data-sets on customized pathway maps by ProMeTra – aeration-dependent gene expression and metabolism of Corynebacterium glutamicum as an example. *BMC Syst. Biol.* **3**, 82 (2009).
42. Kincaid, R., Kuchinsky, A. & Creech, M. VistaClara: an expression browser plug-in for Cytoscape. *Bioinformatics* **24**, 2112–2114 (2008).
43. Lee, H.K., Hsu, A.K., Sajdak, J., Qin, J. & Pavlidis, P. Coexpression analysis of human genes across many microarray data sets. *Genome Res.* **14**, 1085–1094 (2004).
44. Dunn, W.B. & Ellis, D.I. Metabolomics: current analytical platforms and methodologies. *TrAC Trends Anal. Chem.* **24**, 285–294 (2005).
45. Scholz, M. & Fiehn, O. Setup X – a public study design database for metabolomic projects. *Pac. Symp. Biocomput.* 169–180, doi:10.1142/9789812772435_0017 (2007).
46. Thimm, O. *et al.* MAPMAN: a user-driven tool to display genomics data sets onto diagrams of metabolic pathways and other biological processes. *Plant J.* **37**, 914–939 (2004).
47. Paley, S.M. & Karp, P.D. The Pathway Tools cellular overview diagram and omics viewer. *Nucleic Acids Res.* **34**, 3771–3778 (2006).
48. Tokimatsu, T. *et al.* KaPPA-view: a web-based analysis tool for integration of transcript and metabolite data on plant metabolic pathway maps. *Plant Physiol.* **138**, 1289–1300 (2005).
49. Mlecnik, B. *et al.* PathwayExplorer: web service for visualizing high-throughput expression data on biological pathways. *Nucleic Acids Res.* **33** (web server issue), W633–W637 (2005)
50. Caspi, R. *et al.* The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic Acids Res.* **36** (database issue), D623–D631 (2008).
51. Funahashi, A. *et al.* CellDesigner 3.5: a versatile modeling tool for biochemical networks. *Proc. IEEE* **96**, 1254–1265 (2008).
52. Sauro, H.M. *et al.* Next generation simulation tools: the Systems Biology Workbench and BioSPICE integration. *OMICS* **7**, 355–372 (2003).
53. Demir, E. *et al.* PATIKA: an integrated visual environment for collaborative construction and analysis of cellular pathways. *Bioinformatics* **18**, 996–1003 (2002).
54. Schreiber, F., Dwyer, T., Marriott, K. & Wybrow, M. A generic algorithm for layout of biological networks. *BMC Bioinformatics* **10**, 375 (2009).
55. Thomas, J.J. & Cook, K.A. *Illuminating the Path: The Research and Development Agenda for Visual Analytics.* (National Visual Analytics Center & IEEE, Richland, Washington, USA, 2005).

56. Saraiya, P., North, C. & Duca, K. An Insight-based methodology for evaluating bioinformatics visualizations. *IEEE Trans. Vis. Comput. Graph.* **11**, 443–456 (2005).
57. Dwyer, T., Koren, Y. & Marriott, K. IPSEP-COLA: an incremental procedure for separation constraint layout of graphs. *IEEE Trans. Vis. Comput. Graph.* **12**, 821–828 (2006).
58. Dwyer, T. *et al.* Exploration of networks using overview+detail with constraint-based cooperative layout. *IEEE Trans. Vis. Comput. Graph.* **14**, 1293–1300 (2008).
59. Viégas, F.B., Wattenberg, M., van Ham, F., Kriss, J. & McKeon, M. ManyEyes: a site for visualization at internet scale. *IEEE Trans. Vis. Comput. Graph.* **13**, 1121–1128 (2007).
60. Heer, J., Viégas, F.B. & Wattenberg, M. Voyagers and voyeurs: supporting asynchronous collaborative information visualization. in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'07)* 1029–1038 (ACM, New York, 2007).
61. Heer, J. & Agrawala, M. Design considerations for collaborative visual analytics. *Inf. Vis.* **7**, 49–62 (2008).
62. Pico, A.R. *et al.* WikiPathways: pathway editing for the people. *PLoS Biol.* **6**, e184 (2008).
63. Pavlopoulos, G.A. *et al.* Arena3D: visualization of biological networks in 3D. *BMC Syst. Biol.* **2**, 104 (2008).
64. Ball, R. & North, C. Realizing embodied interaction for visual analytics through large displays. *Comput. Graph.* **31**, 380–400 (2007).
65. Berman, H., Henrick, K. & Nakamura, H. Announcing the worldwide Protein Data Bank. *Nat. Struct. Biol.* **10**, 980 (2003).
66. Hermjakob, H. *et al.* The HUPO PSI's molecular interaction format–a community standard for the representation of protein interaction data. *Nat. Biotechnol.* **22**, 177–183 (2004).
67. Hucka, M. *et al.* The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* **19**, 524–531 (2003).
68. Lloyd, C.M., Halstead, M.D. & Nielsen, P.F. CellML: its future, present and past. *Prog. Biophys. Mol. Biol.* **85**, 433–450 (2004).
69. Le Novère, N. *et al.* The Systems Biology Graphical Notation. *Nat. Biotechnol.* **27**, 735–741 (2009).
This publication marks the first serious attempt to create a community standard for a graphical notation to represent networks in systems biology.
70. Walter, T. *et al.* Visualization of image data from cells to organisms. *Nat. Methods* **7**, S26–S40 (2010).
71. O'Donoghue, S.I. *et al.* Visualization of macromolecular structures. *Nat. Methods* **7**, S42–S55 (2010).
72. Nielsen, C.B., Cantor, M., Dubchak, I., Gordon, D. & Wang, T. Visualizing genomes: techniques and challenges. *Nat. Methods* **7**, S5–S15 (2010).
73. Procter, J.B. *et al.* Visualization of multiple alignments, phylogenies and gene family evolution. *Nat. Methods* **7**, S16–S25 (2010).
74. Burrage, K., Hood, L. & Ragan, M.A. Advanced computing for systems biology. *Brief. Bioinform.* **7**, 390–398 (2006).
75. Awad, I.A., Rees, C.A., Hernandez-Boussard, T., Ball, C.A. & Sherlock, G. Caryoscope: an open source Java application for viewing microarray data in a genomic context. *BMC Bioinformatics* **5**, 151 (2004).
76. Lau, C. *et al.* Exploration and visualization of gene expression with neuroanatomy in the adult mouse brain. *BMC Bioinformatics* **9**, 153 (2008).
77. Weber, G.H. *et al.* Visual exploration of three-dimensional gene expression using physical views and linked abstract views. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **6**, 296–309 (2009).
78. Barsky, A., Gardy, J.L., Hancock, R.E. & Munzner, T. Cerebral: a Cytoscape plugin for layout of and interaction with biological networks using subcellular localization annotation. *Bioinformatics* **23**, 1040–1042 (2007).
79. Mookherjee, N. *et al.* Modulation of the TLR-mediated inflammatory response by the endogenous human host defense peptide LL-37. *J. Immunol.* **176**, 2455–2464 (2006).
80. Spellman, P.T. *et al.* Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization. *Mol. Biol. Cell* **9**, 3273–3297 (1998).
81. Kuntzer, J. *et al.* BNDB – the Biochemical Network Database. *BMC Bioinformatics* **8**, 367 (2007).
82. Baitaluk, M., Sedova, M., Ray, A. & Gupta, A. BiologicalNetworks: visualization and analysis tool for systems biology. *Nucleic Acids Res.* **34** (web server issue), W466–W471 (2006).
83. Cline, M.S. *et al.* Integration of biological networks and gene expression data using Cytoscape. *Nat. Protoc.* **2**, 2366–2382 (2007).
84. Hooper, S.D. & Bork, P. Medusa: a simple tool for interaction graph analysis. *Bioinformatics* **21**, 4432–4433 (2005).
85. Kao, H.L. & Gunsalus, K.C. Browsing multidimensional molecular networks with the generic network browser (N-Browse). *Curr. Protoc. Bioinformatics* **9**, 9.11.1–9.11.21 (2008).
86. Brown, K.R. *et al.* NAViGaTOR: Network Analysis, Visualization and Graphing Toronto. *Bioinformatics* **25**, 3327–3329 (2009).
87. Kohler, J. *et al.* Graph-based analysis and visualization of experimental results with ONDEX. *Bioinformatics* **22**, 1383–1390 (2006).
88. Breitkreutz, B.J., Stark, C. & Tyers, M. Osprey: a network visualization system. *Genome Biol.* **4**, R22 (2003).
89. Batagelj, V. & Mrvar, A. Pajek – Program for large network analysis. *Connections* **21**, 47–57 (1998).
90. Forman, J.J., Clemons, P.A., Schreiber, S.L. & Haggarty, S.J. SpectralNET–an application for spectral graph analysis and visualization. *BMC Bioinformatics* **6**, 260 (2005).
91. Auber, D. A huge graph visualization framework. in *Graph Drawing Software* (eds. Mutzel, P. & Jünger, M.) 105–126 (Springer, Heidelberg, Germany, 2004).
92. Zinovyev, A., Viara, E., Calzone, L. & Barillot, E. BiNoM: a Cytoscape plugin for manipulating and analyzing biological networks. *Bioinformatics* **24**, 876–877 (2008).
93. Huttenhower, C., Mehmood, S.O. & Troyanskaya, O.G. Graphle: interactive exploration of large, dense graphs. *BMC Bioinformatics* **10**, 417 (2009).
94. Longabaugh, W.J., Davidson, E.H. & Bolouri, H. Visualization, documentation, analysis, and communication of large-scale gene regulatory networks. *Biochim. Biophys. Acta* **1789**, 363–374 (2009).
95. Streit, M., Lex, A., Kalkusch, M., Zatloukal, K. & Schmalstieg, D. Caleydo: connecting pathways and gene expression. *Bioinformatics* **25**, 2760–2761 (2009).
96. Okuda, S. *et al.* KEGG atlas mapping for global analysis of metabolic pathways. *Nucleic Acids Res.* **36** (web server issue), W423–W426 (2008).
97. van Iersel, M.P. *et al.* Presenting and exploring biological pathways with PathVisio. BMC *Bioinformatics* **9**, 399 (2008).
98. Holford, M., Li, N., Nadkarni, P. & Zhao, H. VitaPad: visualization tools for the analysis of pathway data. *Bioinformatics* **21**, 1596–1602 (2005).
99. Chung, H. J., Kim, M., Park, C. H., Kim, J., and Kim, J. H. ArrayXPath: mapping and visualizing microarray gene-expression data with integrated biological pathway resources using scalable vector graphics. *Nucleic Acids Res.* **32** (web server issue), W460–W464 (2004).
100. Weniger, M., Engelmann, J.C. & Schultz, J. Genome Expression Pathway Analysis Tool–analysis and visualization of microarray gene expression data under genomic, proteomic and metabolic context. *BMC Bioinformatics* **8**, 179 (2007).
101. Letunic, I., Yamada, T., Kanehisa, M. & Bork, P. iPath: interactive exploration of biochemical pathways and networks. *Trends Biochem. Sci.* **33**, 101–103 (2008).
102. Karp, P.D., Paley, S. & Romero, P. The Pathway Tools software. *Bioinformatics* **18** (suppl. 1), S225–S232 (2002).
103. Dogrusoz, U. *et al.* PATIKAweb: a Web interface for analyzing biological pathways through advanced querying and visualization. *Bioinformatics* **22**, 374–375 (2006).
104. Santamaria, R., Theron, R. & Quintales, L. BicOverlapper: a tool for bicluster visualization. *Bioinformatics* **24**, 1212–1213 (2008).
105. Goncalves, J.P., Madeira, S.C. & Oliveira, A.L. BiGGEsTS: integrated environment for biclustering analysis of time series gene expression data. *BMC Res Notes* **2**, 124 (2009).
106. Shamir, R. *et al.* EXPANDER–an integrative program suite for microarray data analysis. *BMC Bioinformatics* **6**, 232 (2005).
107. Sturn, A., Quackenbush, J. & Trajanoski, Z. Genesis: cluster analysis of microarray data. *Bioinformatics* **18**, 207–208 (2002).
108. Hibbs, M.A., Dirksen, N.C., Li, K. & Troyanskaya, O.G. Visualization methods for statistical analysis of microarray clusters. *BMC Bioinformatics* **6**, 115 (2005).
109. Seo, J. & Shneiderman, B. Interactively exploring hierarchical clustering results. *Computer* **35**, 80–86 (2002).
110. Saldanha, A.J. Java Treeview–extensible visualization of microarray data. *Bioinformatics* **20**, 3246–3248 (2004).
111. Dietzsch, J., Gehlenborg, N. & Nieselt, K. Mayday – a microarray data analysis workbench. *Bioinformatics* **22**, 1010–1012 (2006).
112. Gehlenborg, N., Dietzsch, J. & Nieselt, K. A framework for visualization of microarray data and integrated meta information. *Inf. Vis.* **4**, 164–175 (2005).
113. Saeed, A.I. *et al.* TM4: a free, open-source system for microarray data management and analysis. *Biotechniques* **34**, 374–378 (2003).
One of the first and still one of the most commonly used applications for the management, analysis and visualization of microarray data.
114. Hochheiser, H., Baehrecke, E.H., Mount, S.M. & Shneiderman, B. Dynamic querying for pattern identification in microarray and genomic data. *Proc. IEEE Multimedia and Expo Int. Conf.* **3**, 453–456 (2003).

115. Kapushesky, M. *et al.* Expression Profiler: next generation–an online platform for analysis of microarray data. *Nucleic Acids Res.* **32** (web server issue), W465–W470 (2004).
116. Reich, M. *et al.* GenePattern 2.0. *Nat. Genet.* **38**, 500–501 (2006).
117. Quackenbush, J. Microarray data normalization and transformation. *Nat. Genet.* **32** (suppl.), 496–501 (2002).
118. Brettschneider, J.. Collin, F., Bolstad, B.M. & Speed, T.P. Quality assessment for short oligonucleotide microarray data. *Technometrics* **50**, 241–264 (2008).
119. Kauffmann, A., Gentleman, R. & Huber, W. ArrayQualityMetrics–a bioconductor package for quality assessment of microarray data. *Bioinformatics* **25**, 415–416 (2009).
120. Morgan, M. *et al.* ShortRead: a bioconductor package for input, quality assessment and exploration of high-throughput sequence data. *Bioinformatics* **25**, 2607–2608 (2009).
121. Robinson, M.D., McCarthy, D.J. & Smyth, G.K. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26**, 139–140 (2010).
122. Smyth, G.K., Yang, Y.H. & Speed, T. Statistical issues in cDNA microarray data analysis. *Methods Mol. Biol.* **224**, 111–136 (2003).
123. Nesvizhskii, A.I., Vitek, O. & Aebersold, R. Analysis and validation of proteomic data generated by tandem mass spectrometry. *Nat. Methods* **4**, 787–797 (2007).
124. Perkins, D.N., Pappin, D.J., Creasy, D.M. & Cottrell, J.S. Probability-based protein identification by searching sequence databases using mass spectrometry data. *Electrophoresis* **20**, 3551–3567 (1999).
125. Li, X.J. *et al.* A tool to visualize and evaluate data obtained by liquid chromatography-electrospray ionization-mass spectrometry. *Anal. Chem.* **76**, 3856–3860 (2004).
126. Sturm, M. & Kohlbacher, O. TOPPView: an open-source viewer for mass spectrometry data. *J. Proteome Res.* 8, 3760–3763 (2009).
127. Kopka, J. Current challenges and developments in GC-MS based metabolite profiling technology. *J. Biotechnol.* **124**, 312–322 (2006).
128. Broeckling, C.D. *et al.* Metabolic profiling of *Medicago truncatula* cell cultures reveals the effects of biotic and abiotic elicitors on metabolism. *J. Exp. Bot.* **56**, 323–336 (2005).
129. Beckonert, O. *et al.* Metabolic profiling, metabolomic and metabonomic procedures for NMR spectroscopy of urine, plasma, serum and tissue extracts. *Nat. Protoc.* **2**, 2692–2703 (2007).
130. Lindon, J.C. & Nicholson, J.K. Spectroscopic and statistical techniques for information recovery in metabonomics and metabolomics. *Annu. Rev. Anal. Chem.* **1**, 45–69 (2008).
131. Xia, J., Bjorndahl, T.C., Tang, P. & Wishart, D.S. MetaboMiner–semi-automated identification of metabolites from 2D NMR spectra of complex biofluids. *BMC Bioinformatics* **9**, 507 (2008).
132. Hotelling, H. Analysis of complex statistical variables into principal components. *J. Educ. Psychol.* **24**, 417–441 (1933).
133. Kruskal, J. Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. *Psychometrika* **29**, 1–26 (1964).
134. Venna, J. & Kaski, S. Comparison of visualization methods for an atlas of gene expression data sets. *Inf. Vis.* **6**, 139–154 (2007).
135. Inselberg, A. The plane with parallel coordinates. *Vis. Comput.* **1**, 69–91 (1985).
136. Eisen, M.B., Spellman, P.T., Brown, P.O. & Botstein, D. Cluster analysis and display of genome-wide expression patterns. *Proc. Natl. Acad. Sci. USA* **95**, 14863–14868 (1998).
Milestone publication that introduced the heat map visualization to the field of transcriptomics and has been cited several thousand times.
137. Wilkinson, L. & Friendly, M. The history of the cluster heat map. *Am. Stat.* **63**, 179–184 (2009).
138. Weinstein, J.N. Biochemistry. a postgenomic visual icon. *Science* **319**, 1772–1773 (2008).

frontiers in PHYSIOLOGY

OPINION ARTICLE
published: 07 May 2010
doi: 10.3389/fphys.2010.00003

Grand challenges in systems physiology

Hiroaki Kitano[1,2,3]*

[1] The Systems Biology Institute, Tokyo, Japan
[2] Sony Computer Science Laboratories, Inc. Tokyo, Japan
[3] Okinawa Institute of Science and Technology, Okinawa, Japan
*Correspondence: kitano@sbi.jp

Systems physiology is an integrated discipline. It combines experimental, computational, and theoretical studies to advance our understanding of the physiology of human and other living creatures. In other words, systems physiology is systems biology with a physiology (i.e., functionally)-centered view. Understanding the principle behind the system is one of the fundamental challenges in systems physiology and systems biology. One can not make the use of sophisticated computational models or arrays of biological data to deepen our understanding of biological function without in-depth insights into the systems as a whole. For example, robustness and its trade-offs have been proposed as a fundamental principles (Kitano, 2004, 2007). This view, although still speculative, provides a framework for the conceptualization of data and observed phenomena. Identification of a series of such principles and their relationships can enrich our understanding of biological systems. The beauty of a good theory is that it reshapes our view of the world, so that the same data and phenomena may be re-interpreted in the light of the introduced concepts. Such transformation of our conceptualization often leads to true advances in science.

While such theoretical and explorative research are expected, it is also important to consolidate various efforts to achieve high impact objectives; these efforts are often referred as "Grand Challenges." Defining grand challenges provide an effective approach that both illuminates unresolved issues and helps focus research effort on these problems and thereby advances the state-of-the-art in systems physiology. It is most effective when used for engineering-oriented projects where progress can be made by the effective coordination of research and development programs along with a series of technological innovations, rather than merely waiting for serendipitous explorations. While basic scientific explorations are still indispensable and much needed in this field, it is also true that coordinated efforts on relatively well-defined missions can dramatically change the way we do science and apply it to medical practice. In this article, I attempt to define a series of grand challenges that are interlinked and designed to accomplish the ultimate goal of creating an integrated understanding and platform for human healthcare services, biomedical research, and drug discovery.

The grand challenge is to create highly accurate and broad coverage computational model of organisms that are backed up by well-controlled high precision experimental data. In practice, the true challenge is not only to build such a model, but also to establish a system of technologies that enable us to build these models cost-effectively, because these models must match genetic and epigenetic diversity. With this technology, both virtual human and virtual mouse models shall be developed. In addition, models of specific cell lines shall be developed. This set of models shall be consistent with a set of cells and organisms used for drug discovery and biomedical research. The reality of the drug discovery pipeline is that it uses cell lines and animal models before moving into clinical trial. Thus, it is important that not only human models, but also mouse and cell line models are developed with an equal level of quality. Accomplishment of this grand challenge will enable us to use computational models and associated experimental verification systems to understand disease mechanisms, and to predict drug efficacy, side effects, and therapeutic strategy outcomes. At a workshop held in Tokyo in February 2008, a group of researchers agreed to initiate a project to create a "virtual human" in next 30 years (Jones, 2008). They also announced the Tokyo Declaration that reads in part as follows: "Recent advances in Systems Biology indicate that the time is now ripe to initiate a grand challenge project to create over the next 30 years a comprehensive, molecules-based, multi-scale, computational model of the human ('the virtual human'), capable of simulating and predicting, with a reasonable degree of accuracy, the consequences of most of the perturbations that are relevant to healthcare."[1]

Although creation of a virtual human (a comprehensive computational model of human being) has been the subject of much discussion in variety of conferences and workshops, the real implications and difficulties with the model need to be re-addressed. There is no doubt that simulation, if properly used, can be a powerful tool for scientific and engineering research. Modern aircrafts cannot be developed without help of computational fluid dynamics (CFD). CFD is one of the most successful computational approaches used in the engineering design process.

There are three major reasons why CFD is now widely accepted. First, the Navier-Storkes equation has been well established to provide a computational basis for fluid dynamics with reasonable accuracy. While there are yet unresolved issues on how to compute tabular flows accurately, the Navier-Storkes equation provides an acceptable practical solution for most needs. Second, many CFD results are compared and calibrated against wind-tunnel experiments that are highly controlled and extensively monitored. Due to the existence of the wind-tunnel, CFD models can be improved for their accuracy and reliability of predictions. In wind tunnels, air flow speed, temperature, and other parameters can be adjusted within a very small error margin, for example within 0.01% error margin. Third, decades of effort have been spent on improving CFD and related fluid dynamics research. Thus, the current status of CFD is a result of decades of effort.

For computer simulation and analysis in biology to parallel the success of CFD, it must establish a fundamental computing paradigm comparable to the Navier-Storkes equation, to create a wind-tunnel equivalent for biological

[1] http://www.systems-biology.org/~myukiko/FC-SB2008/doku.php?id=workshop:statement

experiments, and to maintain a constant focus on these problems for decades. Of course, the biological system is much more heterogeneous and complex than fluids, but a set of basic equations must be established so that fundamental principles behind the computations point in the right direction. It is essential that both interaction networks and the physical structures are modeled together so that the resulting model provides an improved reality, particularly for high-resolution modeling of complex mammalian cells. Second, highly controlled and high-precision experimental systems that will serve as the "wind-tunnel" in biology are essential. Microfluidics and other emerging technologies may provide us with experimental paradigms that have remarkably high precision (Balagadde et al., 2005).

Even if technologies can be developed, their full potential can not be reached unless they are used properly. There are, at least, two issues that must be carefully examined in order to make the best use of a computational simulation. First, the purpose of simulation has to be well defined, and model has to be constructed to maximize the purpose of the simulation. This affects the choice of modeling technique, levels of abstractions, the scope of modeling, and parameters to be varied. Second, the simulation needs to be well placed in the context of the whole analysis procedure. In most cases, simulation is not the only methods of analysis. Thus, the part of analysis that uses numerical simulation and the other parts that use non-simulation methods must be well coordinated in order to maximize overall success of the analysis activity. An example from racing car design illustrates these issues. CFD is extensively used in Formula-1 car design in order to obtain optimal aerodynamics (Ziemelis and Wenz, 2004); that is, a higher downward force coupled with a lower drag. Particular interest has been placed on effects of various aerodynamics components such as front wings, rear wings, and ground effects. However, the complicated interference between front wings, suspension members, wheels, and break air intake ducts must also be investigated. Combustion in engine is the other issue where simulation studies are often used, but simulated separately from CFD model. This exemplifies the practice of proper focus and abstraction. Thus, one can infer from this example that attempts to create a computational model of the human being without defining the model's use cases and the expected insights to be gained from the model would be economically inefficient and unlikely to be successful.

It should also be noted that CFD is not the only tool used for aerodynamics design. F-1 racing cars are initially designed using CFD (*in silico*), then further investigated using wind tunnel (*in physico*), followed by actual run at the test course (*in vitro*) before being deployed in actual races (*in vivo*). CFD in this case is used for initial search of candidate designs that are subject of further investigation and modification based upon results obtained from wind tunnel testing. This sequence of computational design followed by physical testing (experimentation) is the key for success in engineering design. It is highly likely the same would be true for biology. If so, a series of corresponding experimental platforms and methodologies may need to be developed to make the best use of the results obtained from the computational modeling approach.

Looking at the modeling platform, there are three major issues: scaling, sharing, and merging of biological models. The scaling problem, in turn, has three aspects: problem scaling, layer scaling, and scope scaling. Problem scaling means that the approach or computing framework enable models to get larger and larger to cover a substantial part of the organism. Developing a large-scale model is beyond the scope and capability of a single laboratory, and, in fact, may not even be possible within a national framework. It is critically important to establish an international collaborative framework to provide the infrastructure necessary to supports these activities in order to develop large scale models. This issue is directly related to the issue of sharing and merging of models. This requires installation of platform that fosters a global initiative. For example, there must be a mechanism by which the multiple models developed by different research groups can be combined into a single consistent model. Of course, an underlying assumption is that the models can be shared, requires well-informed communication within the community and the establishment of standards for models as seen in SBML (Hucka et al., 2003)[2] and SBGN (Le Novere et al., 2009)[3].

Layer scaling means that the model can incorporate multiple layers of description from the sub-cellular, cellular, and tissue level to the whole organism and assembly of organisms. This is a non-trivial issue because each layer may have different modalities of operation and a suitable way to represent these layers into models in a consistent and integrated manner is yet to be understood.

Finally, scope scalability can be defined as the capability of modeling approach to allow for an integrated treatment of both interactions between the layers and the physical structures (Kitano, 2006). While many models often used in systems biology focus on molecular interactions and gene regulatory networks, they often neglect the important structures and dynamics of intracellular and intercellular systems as well as the whole body. This is especially the case for physiological studies. For example, models that combine cytoskeleton dynamics, hence cell deformation and movements, with molecular and genetic interactions are at best rare, but more typically totally lacking.

It must be emphasized that one must first clearly establish what scientific questions are to be answered by using computational approach before the model of the biology system is developed. While this criterion has been already stated, it is so critical for the success but all too frequently forgotten during the course of model development that I shall repeat the point again. Mere attempts to create computational models that behave like an actual cell and organisms does not in themselves constitute a good scientific practice. It must be remembered what simulation and modeling represent is an abstraction of the actual phenomena. Without first carefully framing the scientific questions, a proper determination of the right level of abstraction and the scope of the model to be created is not possible. This is also the case in CFD. CFD as used in racing car design has a clear and an explicit optimization goal, namely to maximize the downward force while minimizing drag. The problem for biological simulations is that the information to be discovered by the simulation is much more complex than needed for racing car design. However, the questions must be a well-defined one in order to make the best

[2]http://www.sbml.org/

[3]http://www.sbgn.org/

use of computational machinery. With the right question and framing of problem, the model can become the starting point for a broad range of applications.

The discussion as presented so far outlines a new set of problems and challenges that has not been common in traditional biomedical sciences. This fact may have implications on how scientific communications, including journals, need to be organized and directed. For example, models and other resources that are gaining more importance have not been properly credited. There needs to be mechanisms by which proper credit can be assigned to the large groups of international experts who contribute to the incremental improvement of existing models and other knowledge resources. The proper consolidation of knowledge is as equally important for scientific advancement as are novel discoveries because the simple assembly of isolated knowledge, regardless of originality of discovery itself, does not enable us to achieve the grand challenge.

In order to resolve this issue, this journal attempts to provide a forum for the publication of modeling and mapping studies, results that have often been difficult to publish. The development of precision models, molecular interactions maps, and other knowledge-intensive resources are critical for the advancement of systems physiology and systems biology. In the past, the value of submissions describing these results have not been fully appreciated due to the assumption that these studies fail to provide novel insights despite the fact that important insight is often gained by efficient use of these models and maps. Just like an engineer who can design and build a great car without being a great driver, those who develop precision models and maps can provide functional insights that others can experimentally confirm or refute. In a similar fashion, being a great driver does not means that one can design and build a great car. Thus, someone who can gain insights from the models and maps may not be the one who can actually develop the resources used in the model. Model and map construction are engineering and infrastructure work that requires specific skills and dedication that provide resources essential for promoting systems physiology and such contributions need to be properly credited. A major goal of this journal is to establish an innovative forum of scientific exchange in the new area of web-based scientific activity.

A grand challenge for systems physiology entails this exciting objective. We need a series of innovations, discoveries, collaborative efforts, and dedications to accomplish it. The impact will be massive.

REFERENCES

Balagadde, F. K., You, L., Hansen, C. L., Arnold, F. H., and Quake, S. R. (2005). Long-term monitoring of bacteria undergoing programmed population control in a microchemostat. *Science* 309, 137–140.

Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., Arkin, A. P., Bornstein, B. J., Bray, D., Cornish-Bowden, A., Cuellar, A. A., Dronov, S., Gilles, E. D., Ginkel, M., Gor, V., Goryanin, I. I., Hedley, W. J., Hodgman, T. C., Hofmeyr, J.-H., Hunter, P. J., Juty, N. S., Kasberger, J. L., Kremling, A., Kummer, U., Le Novere, N., Loew, L. M., Lucio, D., Mendes, P., Minch, E., Mjolsness, E. D., Nakayama, Y., Nelson, M. R., Nielsen, P. F., Sakurada, T., Schaff, J. C., Shapiro, B. E., Shimizu, T. S., Spence, H. D., Stelling, J., Takahashi, K., Tomita, M., Wagner, J., and Wang, J. (2003). The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics* 19, 524–531.

Jones, D. (2008). All Systems Go. *Nat. Rev. Drug Discov.* 7, 128–129.

Kitano, H. (2004). Biological robustness. *Nat. Rev. Genet.* 5, 826–837.

Kitano, H. (2006). Computational cellular dynamics: a network–physics integral. *Nat. Rev. Mol. Cell Biol.* 7, 163.

Kitano, H. (2007). Towards a theory of biological robustness. *Mol. Syst. Biol.* 3, 137.

Le Novere, N., Hucka, M., Mi, H., Moodie, S., Schreiber, F., Sorokin, A., Demir, E., Wegner, K., Aladjem, M. I., Wimalaratne, S. M., Bergman, F. T., Gauges, R., Ghazal, P., Kawaji, H., Li, L., Matsuoka, Y., Villéger, A., Boyd, S. E., Calzone, L., Courtot, M., Dogrusoz, U., Freeman, T. C., Funahashi, A., Ghosh, S., Jouraku, A., Kim, S., Kolpakov, F., Luna, A., Sahle, S., Schmidt, E., Watterson, S., Wu, G., Goryanin, I., Kell, D. B., Sander, C., Sauro, H., Snoep, J. L., Kohn, K., and Kitano, H. (2009). The systems biology graphical notation. *Nat. Biotechnol.* 27, 735–741.

Ziemelis, K., and Wenz, C. (2004). Formula 1 racing: Science in the FAST LANE. *Nature* 431, 736–738.

Received: 28 March 2010; accepted: 31 March 2010; published online: 07 May 2010.

Citation: (2010) Kitano H. Grand challenges in systems physiology. Front. Physio. 1:3. doi: 10.3389/fphys.2010.00003

This article was submitted to Frontiers in Systems Physiology, a specialty of Frontiers in Physiology.

Molecular Systems Biology 6; Article number 384; doi:10.1038/msb.2010.40
Citation: *Molecular Systems Biology* 6:384

www.molecularsystemsbiology.com

molecular systems biology

PERSPECTIVE

Violations of robustness trade-offs

Hiroaki Kitano[1,2,3,*]

[1] The Systems Biology Institute, Minato, Tokyo, Japan,
[2] Sony Computer Science Laboratories, Inc., Shinagawa, Tokyo, Japan and
[3] Okinawa Institute of Science and Technology, Kunigami, Okinawa, Japan
* Corresponding author. The Systems Biology Institute, Falcon Building 5F, 5-6-9 Shirokanedai, Minato, Tokyo 108-0071, Japan. Tel.: +81 3 5789 2181; Fax: +81 3 5789 2182; E-mail: kitano@sbi.jp

Received 21.10.09; accepted 12.5.10

Biological robustness is a principle that may shed light on system-level characteristics of biological systems. One intriguing aspect of the concept of biological robustness is the possible existence of intrinsic trade-offs among robustness, fragility, performance, and so on. At the same time, whether such trade-offs hold regardless of the situation or hold only under specific conditions warrants careful investigation. In this paper, we reassess this concept and argue that biological robustness may hold only when a system is sufficiently optimized and that it may not be conserved when there is room for optimization in its design. Several testable predictions and implications for cell culture experiments are presented.
Molecular Systems Biology 6: 384; published online 22 June 2010; doi:10.1038/msb.2010.40
Subject Categories: metabolic and regulatory networks
Keywords: evolution; portfolio selection; robustness; suboptimality; trade-offs

Introduction

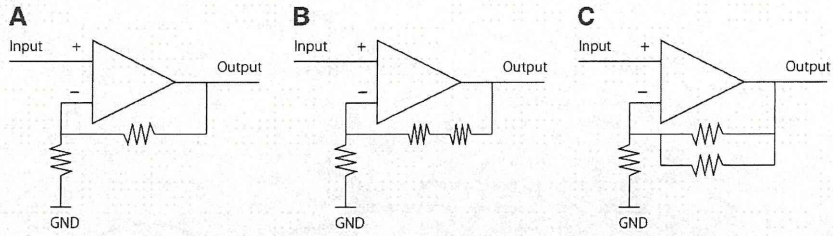
It has been claimed that trade-offs exist between robustness, fragility, performance, and resource demands in biological and engineering systems (Csete and Doyle, 2002; Kitano, 2004, 2007). Determination of the conditions in which this conjecture would hold is of great interest for systems theory in biology. For example, systems that are optimized to be robust against certain perturbations are often extremely fragile against unexpected perturbations. This trade-off is also known as the 'robust yet fragile' nature of highly optimized systems (Csete and Doyle, 2002). Principles such as the Bode integral formula (Bode, 1945) and the summation theorem in metabolic control analysis (Fell, 1997) underscore this trade-off in certain conditions. Although such theorems provide a basis for understanding robustness trade-offs, their applications are limited to specific aspects of the system. The Bode integral formula applies specifically to conservation of the sensitivities of negative feedback circuits on a frequency axis, and the summation theorem assumes linearization for minor perturbations. In addition, real systems are likely to exhibit more complex robustness–fragility trade-offs because of the involvement of component failures and other aspects not taken into account for these theorems (see Supplementary information). Such trade-offs may thus hold only when design and implementation are sufficiently optimized. This means that the system can be made more robust without undermining other features (see Box 1). It should be noted that the system can be optimized by redesign and reimplementation of engineering systems. In biological systems, an evolutionary selection is required for such design optimizations. Although qualitative observations exist for this trade-off, quantitative experimental verification of this trade-off has not been conducted.

In contrast, the trade-off between robustness and performance is more tractable, and several experimental and computational reports discussing such a trade-off have been published (Ibarra *et al*, 2002; Stelling *et al*, 2002; Fischer and Sauer, 2005; Andersson, 2006). In short, the trade-off dictates that high-performance systems are often more fragile than systems with suboptimal performance. Interestingly, there are studies reporting suboptimal metabolism performance in *Bacillus subtilis* and *Escherichia coli* (Stelling *et al*, 2002; Fischer and Sauer, 2005). If the trade-off holds, metabolic performance has to be kept suboptimal to ensure a certain level of robustness against environmental perturbations.

As such studies observe cultured microorganisms and cells, changes in performance and robustness can be attributed to either of the two scenarios: emergence and rebalancing. The emergence scenario assumes that random mutation gives rise to a genetic subtype that fits the perturbed environment better and that this subtype quickly proliferates in culture. The rebalancing scenario assumes that a specific mutant strain that fits the environment better may already exist in a heterogeneous population even before perturbations are imposed, and that this strain proliferates faster than other strains under the perturbed environment.

It is important to clearly define robustness and adaptation through evolutionary selection. Here, 'robustness' means an individual organism's capability of tolerating external and internal perturbations, such as environmental fluctuations, the addition of drugs, and mutations. Robustness–performance trade-off means that, when two individuals are compared, one is found to be more robust than the other but is outperformed by the other; thus, no individual can be more robust and at the same time exhibit higher performance than others. In general, organisms can be 'optimized' or 'adapted' to a certain condition by evolutionary selection; thus, they can be more robust against perturbations implicated in the

Box 1 Design suboptimality and robustness–fragility trade-offs



The robustness–fragility trade-off is one of the most widely known trade-offs for biological and engineering systems. A simple toy example using electric circuits is presented here to show that the trade-off holds only when system design and implementation are sufficiently optimized. In other words, a circuit can be made more robust without scarifying other features.Take a simple electronics example: Assume a simple feedback amplifier in which a feedback loop consists of resistors (Design **A**). There is clearly a robustness–fragility trade-off due to the feedback loop, as depicted in the Bode theorem (Bode, 1945). Without affecting the trade-off due to the feedback, the actual implementation of the amplifier can be changed to have two serially connected resistors in its feedback loop (Design **B**). Failure of one of these resisters may cause dysfunction in the feedback loop and undermine the system stability. Alternatively, the use of resistors connected in parallel would reduce such a failure risk (Design **C**). The parallel implementation is more robust against component failure than both a single and a serial connection configuration. Assuming that a probability of failure of each resister is P, Design B is susceptive to component failure, as a probability of failure of the feedback loop simply doubles ($2P$). Design A reduces such a risk to half of that of Design B because it only uses one resister; hence, probability of failure is P. Design C improves robustness against component failure on feedback loops because of parallel construction. Now it has only P^2, whereas resource demand is equivalent to Design B. For example, a design change from Design B to Design C improves the robustness of the system against component failures without increasing fragility elsewhere, undermining performance, or requiring major additional resources. Thus, robustness is improved without substantial trade-offs. Change from Design B to A actually reduces resource demands slightly and improves robustness. Change from Design A to C improves robustness with minimum increase in resource demands. However, attempts to totally eliminate component failure, not only for a feedback loop but for every aspect of the system, would require multiple redundancies for every aspect of the system, which would require major resources. How design affects the system vulnerability to component failure is a complex issue, and biological cases need to be further explored. It will be further complicated when feedback loops are involved (see Supplementary information).

selection pressure or can have higher performance than preselected individuals. Thus, if the robustness–performance trade-off holds, descendents of organisms can be more robust than their ancestors when they are adapted for perturbations imposed during evolutionary selection, but they may be outperformed by their ancestors or by other individuals adapted for other conditions in which performance is favored. By the same token, the descendents of organisms can outperform their ancestors when selection pressure favors high-performing individuals, but may be less robust than their ancestors and other individuals evolved under conditions that favor more robust individuals.

In this paper, we examine the idea that such trade-offs appear only when the system is sufficiently optimized, and thus may not be observed when systems are yet to be fully optimized. This implies that there should be cases in which descendents of organisms can be more robust and perform as well as or better than their ancestors, which is not possible if robustness–performance trade-off holds universally.

A primer on portfolio selection

In this article, the portfolio selection concept used in modern investment theory is introduced to explain robustness–performance trade-offs. Portfolio selection is an idea to combine various investment options to minimize risk while attaining the desired return on investment (Markowitz, 1991). In the modern portfolio theory used in investment practice, it is well understood that there is trade-off between risk (uncertainty of the return shown by s.d.) and performance (expected return). High-yield financial products generally have higher risk, and modest-yield products have lower risk. Risk in this context refers to the s.d. of the asset price. Performance is measured by the expected percentage of return. Any investment item (asset) can be mapped on a yield–risk space.

As investors generally invest in multiple financial assets with different expected yields and risks, the question is how to find the optimal mix of assets with a desirable yield and acceptable risk. The concept of efficient frontier needs to be introduced here. The efficient frontier is a set of points that represent an optimal combination of assets (mostly securities in a financial context) that maximizes the return for any given level of s.d. Any point not on the efficient frontier represents a portfolio that is inferior to a portfolio on the efficient frontier, either because it generates less return at the same level of risk or is exposed to higher risk at the same expected level of return. In Figure 1A, Portfolio X is inferior to both Portfolios Y and Z. Portfolio Y has a higher expected return than X at the same level of risk, and Z has lower risk than X with the same expected return. Portfolio X can be reorganized to reach the efficient frontier. Thus, theoretically, any portfolio not on the efficient frontier can improve its yield without increasing risk, or reduce risk without undermining the expected yield. However, on the efficient frontier, any change in yield affects risk and *vice versa*. Trade-off between risk and yield takes place on an optimal portfolio that is on the efficient frontier. A similar trade-off concept is also investigated in the context of multiobjective optimization as the Pareto efficiency, originally proposed by Pareto (1935). For a Pareto-efficient solution, no individual parameter can be improved without undermining another parameter. A set of Pareto-efficient solutions constitute a Pareto surface, also called a Pareto frontier.

An indifference curve projects valuation criteria on the yield–risk space. It has graded utility levels depending on

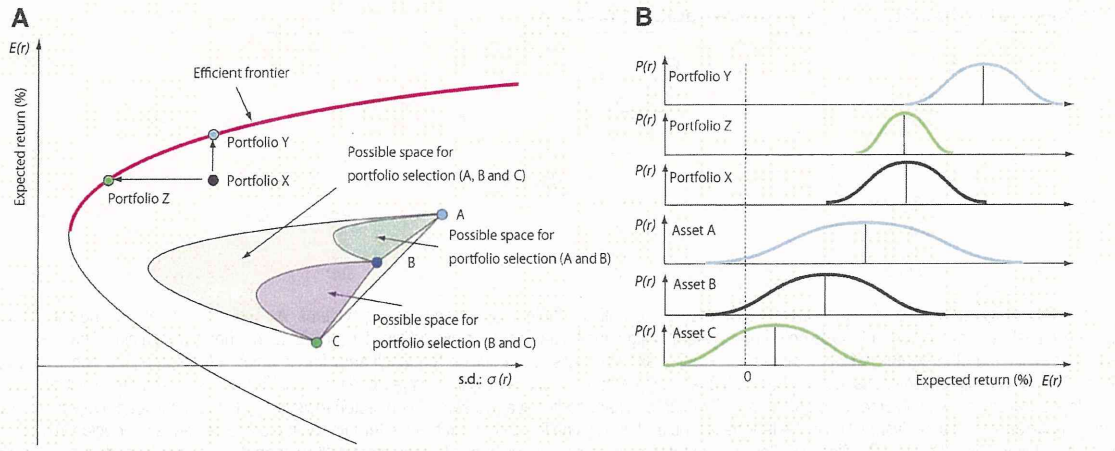


Figure 1 Basic concepts of portfolio selection. (**A**) Any asset can be mapped on the $E(r)$–$\sigma(r)$, or yield–risk, space. Combining A and B creates a possible space of yield and risk depending on the mixture and covariance of the two assets against fluctuation. Maximum risk reduction is achieved when the prices of the two assets change in opposite directions because this offsets fluctuation. Increasing the number of assets involved generally reduces risk. The efficient frontier is achieved by optimally combining all available assets. Availability of a larger number of assets with different yield–risk characteristics improves the overall portfolio, analogous to an increase in the degree of design of freedom in highly optimized tolerance (Carlson and Doyle, 1999; Reynolds *et al*, 2002). In actual investment planning, investment with a fixed return asset is considered to form a capital market line, but this is not considered in biological applications because there is no zero-risk-fixed-yield genotype. (**B**) The probability distribution of expected return is shown for each asset and portfolio in (A) to visually illustrate their relationships.

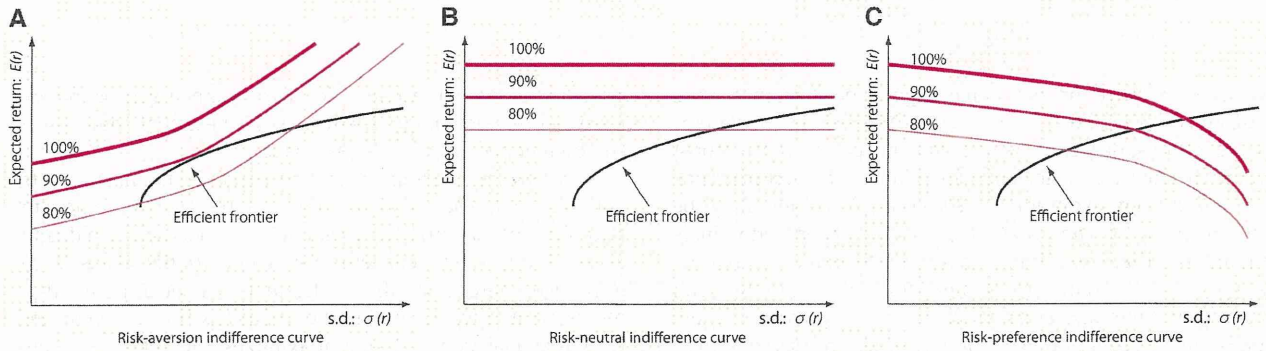


Figure 2 Indifference curves. Three indifference curves are shown: (**A**) risk-aversion, (**B**) risk-neutral, and (**C**) risk-preference curves. Percentile numbers associated with each line indicate the level of utility, hence the level of satisfaction or optimality in the given context.

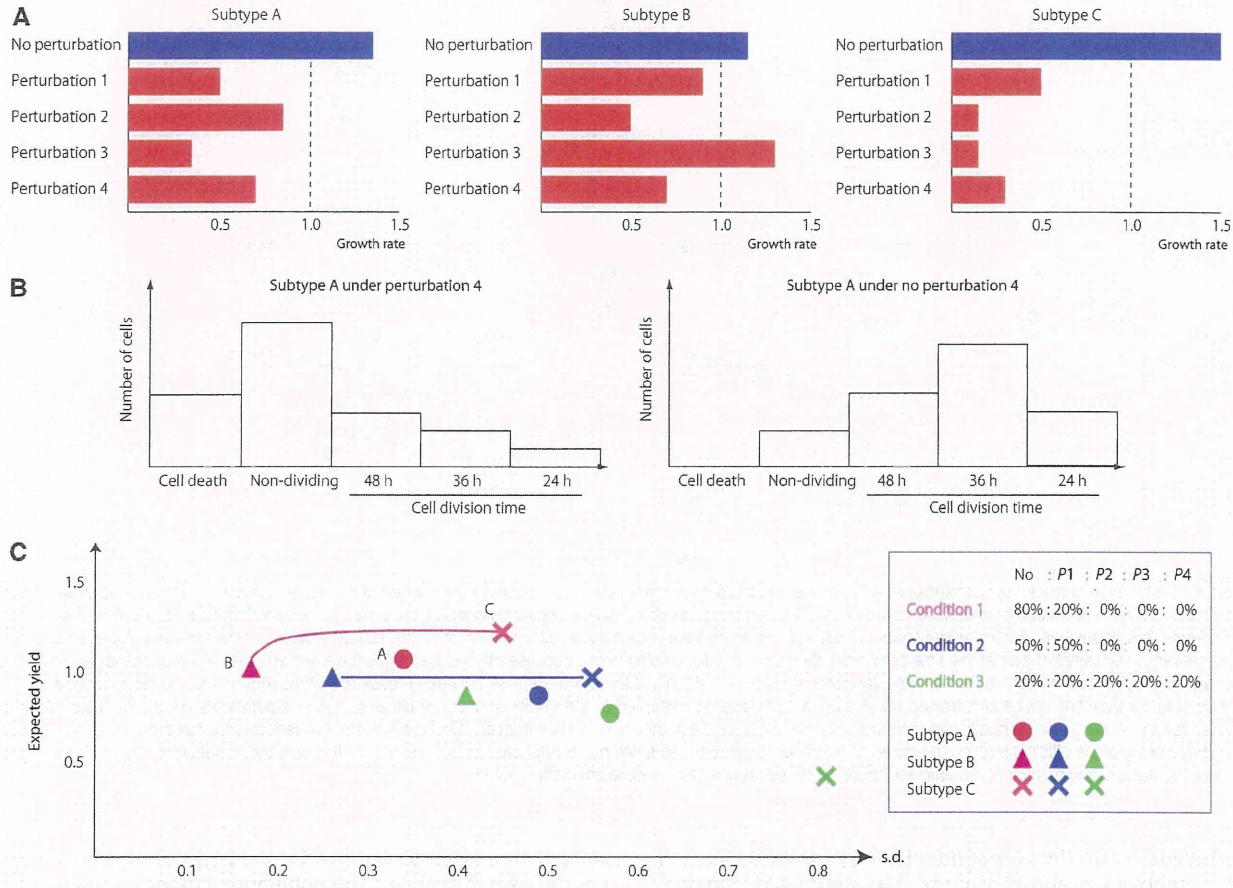
whether the desired portfolio is selected on the basis of risk preference. A risk-aversion indifference curve represents a portfolio chosen to maximize the expected return but avoids risk (Figure 2A). A risk-neutral indifference curve is used when only the expected return is considered (Figure 2B). A risk-preference indifference curve is used when higher risk is preferred for an equal expected return (Figure 2C). Obviously, the risk-preference indifference curve would be an odd choice for an investment situation. Thus, the risk-aversion indifference curve is used in general.

Genetic portfolio selection: translating investment theory into biology

Portfolio selection, which seems remote from biology, can be applied to understand the evolution of microorganisms and cells in specific conditions. Thus, it may help us understand robustness–performance trade-offs.

Each organism or cell can be mapped onto the yield–risk space. A position of the yield–risk space that characterizes the biological entity X can be called 'a projected position of X.' Yield is an expected performance, such as reproduction speed or biomass production rate. Risk (equivalent to fragility) is the degree to which a growth rate or biomass production rate is affected by perturbations. In general, it can be represented by s.d. and calculated by assuming possible perturbations, their probabilities of occurrence, and expected yield under each perturbation (Box 2). These indexes can be measured by tracing behaviors of individual cells and their biomass production or lineage for reproductive efficiency under various conditions. Alternatively, they can be measured by the growth of the population under various perturbations in which the population can be kept monoclonal. In this case, the distribution of projected positions of a certain cell or organism

Box 2 Mapping cell culture onto yield–risk space



Suppose the yield is the growth of an organism. Expected yield and its s.d. can be determined through a series of perturbation experiments. For example, the growth rate of each subtype may be measured under a normal culture condition (no perturbation or reference) and with different perturbations (**A**). Selection is imposed at an individual level. Histograms (**B**) indicate the number of cells that result in cell death, nondividing cells after a certain period of time, and dividing cells with different intervals for subtype A under different conditions. Expected yields in (A) are the results of such responses of individual cells for each condition.Subtypes can be mapped on the yield–risk space as shown in (**C**). Under the condition that 80% of the time there is no perturbation and 20% of the time perturbation 1 occurs, each subtype can be mapped on the yield–risk space as shown by red symbols. In this case, a combination of subtypes B and C reduces risk without major sacrifice in yield. Because of the covariance of B and C that has a very small positive value (0.068), mitigation of risk over the risk of subtype B is limited. The projected position of each subtype in the yield–risk space will be different if assumed perturbations and their probability of occurrence are different. Under the assumption of no perturbation and perturbation 1 conditions, each with a 50% probability, the projected position of each subtype in the yield–risk space is shown by blue symbols, which are located differently from the red symbols that assume no perturbation for 80% probability. The expected yields of subtypes B and C are now equal; thus, a combination of B and C only reduces risk, yield is not reduced. Yet another assumption of perturbations, in which no perturbation and perturbations 1, 2, 3, and 4 each occur with 20% probability, would result in each subtype being located as shown by the green symbols. In this case, subtypes A and C suffer seriously from perturbations; thus, the population will be dominated by subtype B, which has robust yet low-yield characteristics.

for its wild type and various mutants on the yield–risk space is contained by the efficient frontier (Figure 3). Changes in the distribution of projected positions in the yield–risk space for randomly sampled cells will test the conjecture that the robustness–performance trade-off holds only at the efficient frontier.

Next, we consider the cases in which such a trade-off holds in a population of microorganisms and cells. Analysis at the population level is biologically important because cell cultures that have a substantial level of heterogeneity are often used in biological experiments. In addition, certain tumors are known to be composed of heterogeneous cancer cells. Furthermore, populations of organisms and cells are used to measure growth rate and how organisms respond to environmental changes in the context of the study of general biology and in drug screening.

Growth rate (yield or performance) is generally measured by the size of a colony, by numbers of cells, or by other means that reflect the number of cells in the population. Risk is an s.d. of growth rate under various possible perturbations. Experimentally, it can be measured by repeated perturbation experiments. In a heterogeneous cell culture, the projected position of the cell culture in the yield–risk space is determined by the population composition. This is illustrated in Figure 4. Initially, the cell culture is mainly composed of subtypes A and B, with a negligible amount of C (Culture 1 in Figure 4A). Subtype C has a better fit with the culture condition and has

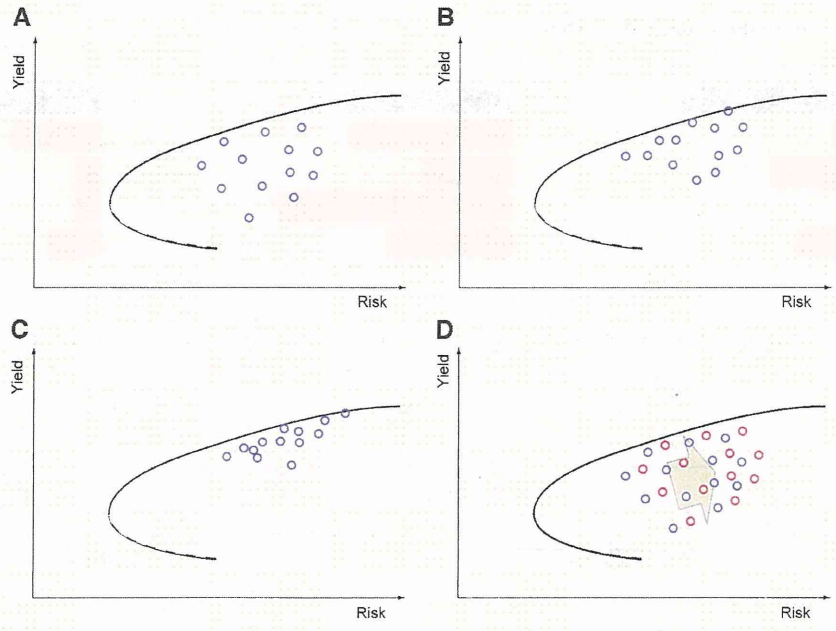


Figure 3 Distribution of randomly sampled cells on yield–risk space. A heterogeneous culture of cells is composed of a mixture of wild type and various mutant cells. (**A**) At the initial stage of culturing, random sampling of cells and mapping onto the yield–risk space may result in a broad distribution for the yield and risk of each cell. (**B**) Culturing this population under the stationary condition for multiple passages may result in evolution of the culture toward a high-yield genotype. Each circle represents randomly sampled cells at this time point, and not the same cell sampled in (A), because multiple passages have occurred. (**C**) Further passages under the stationary condition result in a distribution strongly favoring high-yield individuals. If there is a trade-off between robustness and performance, hence an inverse of risk and yield holds, distribution will be constrained within a certain envelope represented by the efficient frontier. Artificial evolution experiments with random sampling for yield–risk space mapping test the conjecture of robustness–performance trade-off at the efficient frontier. (**D**) If the robustness–performance trade-off holds at any time, even without being at the efficient frontier, the center of gravity of all randomly sampled points after passages will simply shift toward the upper right. This contrasts to the case that the trade-off holds only at the efficient frontier. This difference can be experimentally verified.

higher yield. Thus, the proportion of subtype C increases over that of subtypes A and B (Culture 2 in Figure 4A). Random mutation gives rise to subtype D. It has higher reproductive potential under this specific culture condition, and thus it quickly proliferates in the population (Culture 3 in Figure 4A). However, when extra perturbations are imposed on the culture, fast-growing but less-robust subtypes (subtype D) may substantially decrease in their proliferation speed or the number of cells. At the same time, low-yield but more-robust subtypes may continue to grow at a similar rate. These population changes result in a composition that better fits a condition with a higher degree of perturbations. In this case, the projected position of the culture on the yield–risk space map may move left to that of Culture 4 in Figure 4A. In contrast, the fast-growing subtype may establish its dominance when the environment reaches a more stable condition that is ideal for the fast but less-robust subtype (Culture 5 in Figure 4A). Both emergence and rebalancing scenarios are included, but for the sake of explanation, only the wild type and its mutational variants are used as subtypes. However, cells with different epigenetic modifications can be considered as subtypes if these modifications affect the yield–risk characteristics of the cell.

Although translation of portfolio theory for biology is shown to be possible, some clear and essential differences have to be made explicit and given a new interpretation that is consistent with biology. First, portfolio selection assumes that there are investors and fund managers making decisions regarding the composition of assets selected for the portfolio. This is clearly not the case in biology. The population composition changes because of the relative growth rate of each cell subtype that is the aggregated effect of individual cell reproduction cycles. Individual cells and organisms will be the subject of selection. Second, in portfolio selection theory, investors decide on the indifference curve to be used on the basis of their appetite for risk taking. In the biological translation, indifference curves are only a reflection of the level of perturbation imposed on organisms and cells (Figure 4B). When organisms and cells are cultured in a highly stationary environment, the use of the risk-neutral curve may best predict their possible evolutionary paths. Risk-aversion curves represent the situation in which perturbations are imposed on organisms and cells.

Performance suboptimality and robustness trade-offs

Studies report that suboptimal metabolism performance exists in microorganisms such as *B. subtilis* and *E. coli* (Stelling *et al*, 2002; Fischer and Sauer, 2005). Fischer and Sauer (2005) argue that several regulatory mutants that have improved biomass production efficiency were 'almost exclusively regulators of not-yet-activated adaptive responses, suggesting that *B. subtilis* invests valuable resources in anticipation of changing environmental conditions at the expense of optimal growth'. As almost all mutations to enhance biomass production