To build a better model

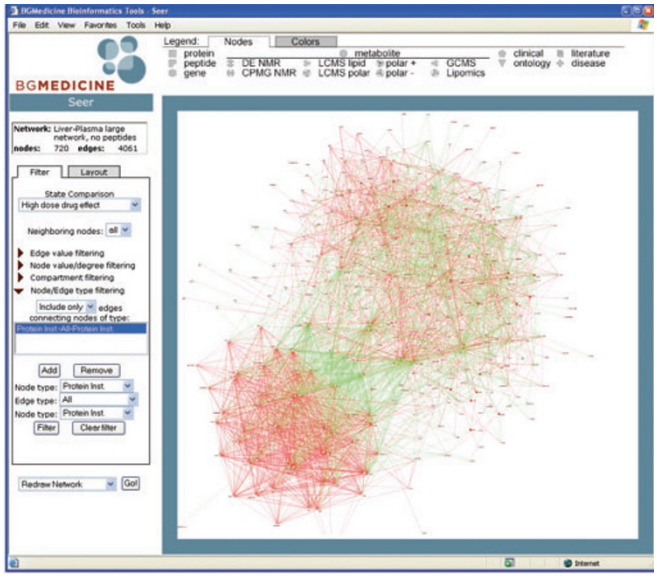
As the gap between the amount of data and the tools for analysis continues to grow, biologists are looking to mathematical modeling to turn data sets into biology. This is bad news for those who studied biology to avoid mathematics—but take heart, the tools are getting better and easier to use, and the best of them are now being used in some inspiring ways. Steven Buckingham reports.

Before the genome age, data were the limiting factor; now, it seems, we have more than we can cope with. Large-scale biology is generating terabytes of new data every week, and biologists in industry and academia alike are beginning to exploit computational methods, such as computer modeling, to make the most out of this mountain of information.

Modeling a complex pathway with the aid of a computer allows us to do what could never be done with traditional intuitive models. Models can show how pathways may be combined in a living system, and can explain how subtle changes in molecular quantities can switch a cell from one state to another or even which part of a large protein network would be the most sensitive target for a new drug.

Access to genomics and proteomics tools has been empowering researchers with the ability to capture snapshots of multitudes of biological components at once, characterizing their state—such as the level of expression of transcripts or the interaction partners of a protein—while the biological system is exposed to various conditions. Generally speaking, researchers still build models based on their understanding of pathways and then feed the models with the enormous data sets characterizing all the pathway parts in the various conditions and extract information to refine the models. The scale of complexity of these data sets, however, has led some to argue for different approaches to model building (see **Box 1**).

Computer models are being used in many practical applications. Entelos Inc., for instance, uses 'virtual patients' for drug discovery and development. They recently completed their Cardiovascular PhysioLab platform, a large-scale computer simulation of cholesterol regulation, atherogenesis and cardiovascular risk. They hope that these comprehensive models will allow them to rapidly assess new drug targets, evaluate combination therapies, identify and interpret biomarker patterns, and predict a drug's long-term clinical efficacy. Entelos is already conducting research in partnership with three big pharmaceutical customers using this platform.



BG Medicine's Seer program is used to explore Correlation Networks constructed using cross-omics data sets. These networks assist in determining drug mechanisms of action, for example. (Courtesy of BG Medicine.)

A lot of the computational modeling work is conducted by academic groups working independently within loose implicitly agreed structures that can help integrate the cumulative efforts. For example, most computer models are built on a common language and stored in public databases. Those models can thus be shared between researchers, allowing published results to be checked and replicated. Chunks of code can also be reused, speeding up the process of generating new models.

But computational modeling has to overcome several challenges if it is to match up with the impact made by established genomic, protein and structural databases. How should a model's data be represented? How should models be stored in a way that allows seamless interoperability? How can model databases be made to work together? What is the minimum annotation that should be stored with the model? These problems have been largely solved in the case of the established databases of biological information and their tools, pointing the way for similar solutions to the challenges facing modeling software.

Languages for modeling

The first challenge faced by the modeling community is in the way the models are described in the first place. Mark-up languages, which are really special cases of XML, the Extensible Markup Language recommended by the World Wide Web Consortium, have established themselves successfully to fill this need and have now reached a stage of maturity. Mark-up languages allow models to be used by different types of software, and provide a systematic, universally agreed way of describing the model's components and how these components fit together. What is more, they make sure that models will outlive the software they were created on. The two most popular mark-up languages are the Systems Biology Markup Language (SBML) and Cell ML.

SBML resulted from the work of Andrew Finney, Herbert Sauro, Hamid Bolouri and Mike Hucka with funding from the Japan Science and Technology Corporation, but rapidly became a community effort. The SBML consortium now meets twice every year, once for language and once for software development.

SBML is updated by the release of new 'levels', in much the same way as software packages come in 'versions'. These levels take the form of specifications and, like software versions, aim to be backwards compatible. Sub-versions have been issued from time to time to meet requests for specific features. Level 1 had only a very small set of features, whereas level 2 saw the replacement of text-based representations with MathML and the introduction of support for metadata. One of the fixes applied to level 2 met the emergent need for stochastic modeling.

Hucka is the only member of the original SBML team still working directly on the project, but his enthusiasm has not slackened one iota. "We already have some great things in store for level 3, and the main goal is to allow models to be more modular, that is, to have a basic set of language features and then to have add-ons. The idea is to allow modular plug-ins to be layered on to level 2." Protein modifications are a case in point: modeling the phosphorylation state of a protein can be done now, but you have to list each state separately. The forthcoming modular design would allow phosphorylation state to be added as an attribute to the molecule, without complicating the models that do not take phosphorylation into account. Hucka also looks forward to the day when models are easier to link with experimental data. "A big area for future of modeling is to make it easier to develop models from the data, such as getting pathways from gene expression data, for instance. There isn't really anything like that at the moment."

Modeling markup languages continue to accumulate features, but how quickly should they change? "Deciding just how quickly the language should evolve has always been a bit of a balancing act," says Hucka. "Many users are demanding new features depending on their research interests—spatial characteristics, for instance. But some users want things to go slowly so they have time to debug and gain experience. So there is a balance here. Nobody is screaming, so we must be getting it about right."

SBML does indeed suffer from the criticism that it has no inherent way of representing spatial information. For instance, the way a molecule is distributed in space, such as a diffusion gradient, could be critical for a cell's behavior. Hucka agrees: "spatial modeling is the next thing SBML needs to get into—some programs do this very well and can export SBML, but SBML doesn't have spatial elements." CellML, in contrast, encapsulates spatial information through leveraging a separate language, FieldML, which provides a set of definitions that describe changes in a value over a field (such as a concentration gradient) or an area (such as in a membrane).

Model repositories

We take it for granted now that when you look up a protein in one of the popular databases maintained by the European Bioinformatics Institute (EBI) or the US National Center for Biotechnology Information (NCBI), there are several links associated with the entry that take you seamlessly into the corresponding entry in different databases. There are even web sites that collate diverse data from diverse

BOX 1 ENGINEERING OR BIOLOGY?

Although models still very much derive from the step-by-step dissection of pathways, not everyone agrees that understanding all the elements in a pathway is the only way to build models. Keith Elliston, CEO of Genstruct, questions whether we really need to know about all the parts, at least for practical purposes. "Our current state of knowledge places us at a crossroads where we need to rethink the way we do science," Elliston argues. "We have rich data sets, but getting hypotheses out of them is where we get stymied."

As an alternative to engineering-style modeling, Genstruct uses an artificial intelligence approach. They accumulate a knowledge base of experimentally validated assertions and use traditional artificial intelligence to suggest hypotheses. For instance, one such assertion might be 'transcription factor *x* upregulates gene *y*'. Then they look at all the genes that this transcription factor can regulate. If there is an effect of a drug, say, on a big proportion of these genes, the system infers that transcription factor *x* may be involved in the response. This mimics the way an expert thinks naturally when doing science: only a computer can do it with so much more data.

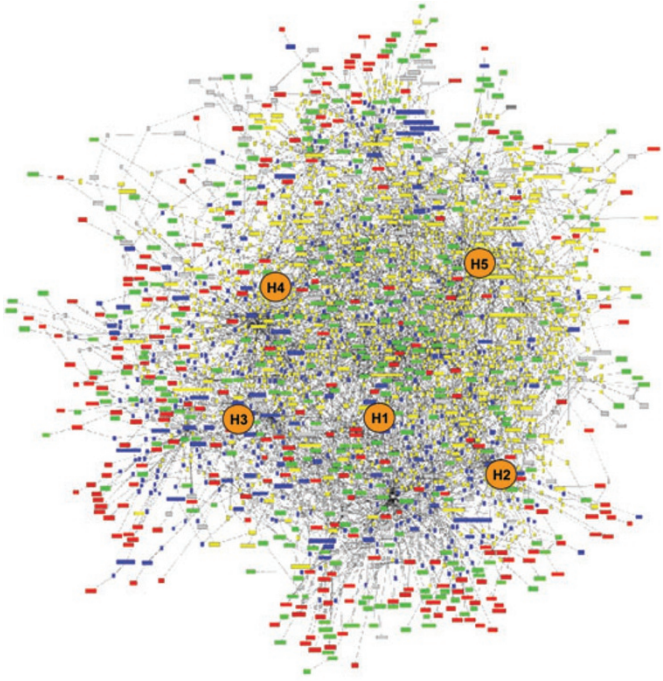
"We see this as a way of brokering the relationship between the experimenter and the computer," says Elliston, who considers their approach to be more biologically relevant than traditional engineering approaches. "Engineer-style modeling would work for understanding a passenger airliner, because it is designed to have one output for each input. Cells are not like that—they have loads of cross-talk and there are many ways to get from A to B." Elliston believes some of the failures in rational drug development are a result of cells being remarkably good at getting around a blockage in a given pathway.

Genego, a company that specializes in providing data-analysis solutions for systems biology, uses another approach altogether to model the effects of drug actions. Like many companies they map 'omics data onto networks of interacting proteins, but where they differ is in the design of their database: instead of being built of molecular species (genes, proteins) it centers on pathways, its basic building blocks being 'elements', 'blocks' and 'entities'. The advantage of this approach is that data of different kinds can be merged. This modeling approach underlies their software products that include MetaCore, which permits the results of high-throughput experiments to be mapped onto networks, and MetaDrug, which allows the effects of drugs on pathways to be evaluated.

databases and collect them cleanly into a common web page. In comparison, model databases lag years behind.

Nicolas Le Novere has been overseeing the development of EBI's BioModels Database (http://www.ebi.ac.uk/biomodels/), which is a repository of models integrated into a database system. Le Novere dreams of the day when you will be able to access a modeling paper in PubMed and follow links straight to the model—and even see the model working through a web interface. Even better, would be the ability to follow links from a protein database entry to find all the models showing that protein working in a network or pathway. "When I look at computational biology tools today, I see an urgent need for three things," says Le Novere, "ways to integrate data from different sources, ways to abstract information from a lower level to a higher one and ways to relate different kinds of information together, like gene expression data with kinetic models, for example."

But researchers will not be able to integrate models until they have an agreed



Cause and effect model of the mechanisms involved in the transition of prostate cancer from androgen dependence to androgen independence (green is observed increase, red is observed decrease, yellow is predicted increase and blue is predicted decrease). (Courtesy of Genstruct.)

system of identifiers—rules about how molecular species are to be named. Users commonly name the model's components (such as proteins) in an idiosyncratic way, often using short, almost cryptic labels. But even a name that is perfectly clear to a biologist cannot be unambiguously identified or linked to a specific protein in a database, let alone understood by a computer program. "If we are ever to see full integration of modeling data, we desperately need to have agreed rules on how to name molecular species," argues Le Novere.

One possible answer is the use of ontologies—strict rules for naming entities and their relationships—which are stirring up a lot of excitement in the field of computational biology and in modeling circles. Until recently, ontologies were deemed too poorly structured to allow computer-based reasoning, but a stream of continual improvements has now resulted in an impressive set of tools, such as the semantic web, open document format (ODF) and the ontology web language (OWL). Tom Plasterer, principal scientist at BG Medicine, thinks the semantic web could be 'the next big thing' in modeling. "Pharma has a lot invested in the federated approach," says Plasterer. "However, if the semantic web gets picked up by the public databases and subsequently by Pharma, the return on investment for adopting this approach could be very attractive, due to much more efficient data and analysis integration both in-house and with partners."

Another lesson learned from the development of existing databases is the need for careful annotation of the data. This has largely been taken care of by the development of minimum information standards: MIAME, for example, is the agreed minimum standard for microarray data. For models, MIRIAM (minimal information requested in the annotation of biochemical models) has just been registered with MIBBI (minimum information for biological and biochemical investigations)—a clearing house for the diverse minimal information specifications in different fields—and it sets out what is agreed as being the acceptable standard of annotations.

"I have been amazed at the increase in the numbers of modeling programs, particularly over the past 2 years," says Mike Hucka.

As more biologists are seeing the possibilities of quantitative computer modeling, databases of biological models are starting to grow. The two best-known databases, BioModels Database and the CellML repository, presently house some 132 and 355 models, respectively. But the number and size of the models being placed in such repositories is making it difficult to keep up with the curation. BioModels Database is dealing with this by allowing models that are syntactically correct but are awaiting to be fully curated, to be stored in a special branch of the repository—a sort of model purgatory.

Douglas Kell, Director of the Manchester Center for Integrative Systems Biology, argues for the adoption of web services by the modeling community.

At first sight it may appear that there are very few differences between the two most popular repositories. The BioModels Database and the CellML repositories both allow the user to view the model as a traditional flow diagram and to download it. You can link out to the PubMed record or e-mail the model creator. BioModels Database contains links to the proteins (but no links back) but the models are not rigidly categorized by pathway type, such as circadian or intracellular signaling. The CellML repository allows models to be imported directly in the PCEnv modeling package. The BioModels Database, however, is a true database rather than a repository, and more than half its models are fully tested and curated. Furthermore, unlike CellML, BioModels Database exports SBML, which is a widely adopted standard. With these subtle but important differences, it is anyone's guess whether one of these will become the predominant repository or whether they will diverge to fill different niches.

Modeling software

As the number of modelers grows, it is not surprising that there has also been a sharp increase recently in the number of modeling tools available. There are about 110 on the SBML web page (http://sbml.org/index.psp), where there were only 50 two years ago. "I have been amazed at the increase in the numbers of modeling programs, particularly over the past 2 years," says Hucka. "There is about one being added every month, and that is just the ones we list at the SBML site." Having so much choice makes it difficult to decide which software meets your own particular needs. Bolouri, one of the founders of SBML, thinks that the explosion of software is not being fully exploited by biologists. "We are seeing an ever widening gap between the data and the ability to exploit the data. The tools are becoming available, but their use is lagging behind."

But there is also a more serious danger: Bolouri fears that using different tools might even produce different results. Hucka has similar worries. "What we don't want is tools silently ignoring things they can't deal with. We need the software to tell us, 'I can't handle that'." To address this problem, Hucka is working on a project to develop a test suite that puts software through a standardized road test. In the meantime, some of these concerns at least can be addressed at the language-specification level. Indeed, CellML metadata, for example, lists which tools were used to run the model and how well the model runs in a specific simulation environment.

Modeling programs differ in the features they offer. The E-Cell program, part of the comprehensive E-Cell Project (http://www.e-cell.org) is notable for its near-perfect hybrid simulations, that is, simulations that use a combination of several algorithms. It has a simple scripting interface that makes creation of events easy and the program has 13 different modeling algorithms, any combination of which can be mixed in a single simulation. It allows real-time user interaction and visualization during the simulation, so you can manually adjust a parameter in the middle of a simulation. It even allows parallel computing, either in the form of distributed computing or parallel stepper scheduling. Another simulation program, Copasi (complex pathway stimulator; http://www.copasi.org), allows parameter scan and optimization as well as highly advanced numerical analysis. It also permits parameter estimation using experimental data. Parameters can be changed interactively using sliders or globally to facilitate changing several kinetic rates at the same time.

The modeling software programs on the SBML website support the language to various extents. The most comprehensive support for SBML is provided by SBMLodeSolver (SOSlib; http://www.tbi.univie.ac.at/~raim/odeSolver/). SOSlib is both a programming library and a set of command-line applications for symbolic and numerical analysis of a system of ordinary differential equations. The present release features basic sensitivity-analysis routines.

Several commercially licensed software suites for pathway analysis are available. Pathway Analytics from Teranode Corporation allows integration of different sorts of data, such as those found in the Kyoto Encyclopedia of Genes and Genomes (KEGG; http://www.genome.jp/kegg/), Systems Biology Markup Language (SBML; http://sbml.org/index.psp) or the EBI (http://www.ebi.ac.uk/), including experimental data, into a pathway which can then easily be transformed into a model for simulation. SimPheny from Genomatica is claimed to be a complete biological knowledge management suite, and touts its ability to integrate different kinds of high-throughput data as the key to bridging computational and biological research. PathwayLab from Innetics allows pathways to be built in an editor based on Microsoft Visio, thereby facilitating integration with Microsoft Office suite. Mathematica, the popular program sold by Wolfram Research, also has an extension for running simulations. Jacobian from Numerica Technology, the fruit of 10 years of research at the Massachusetts Institute of Technology, boasts computational efficiency, interaction through graphical user interfaces, and a powerful application programming interface (API) for interfacing with several languages including C, C++ and Excel.

"We are seeing an ever widening gap between the data and the ability to exploit the data. The tools are becoming available, but their use is lagging behind," says Hamid Bolouri.

Recently, MathWorks extended their well-known MatLab software package with the SimBiology program. SimBiology allows easy pathway construction using a block diagram editor, which it translates into an SBML model. Simulations can be run repetitively (which is useful for stochastic and Monte Carlo simulations) and

permits parameter estimation and sensitivity analysis. Its integration with MatLab makes it easy to integrate user-developed code with the model, something that will be particularly attractive to existing MatLab users. The SimBiology website provides an impressive battery of demonstrations and webinars to help train the new user, and MathWorks provides training seminars at venues around the world, if their impressive documentation were not enough.

Web services—the way forward

Another solution to defuse the dangers of running simulations on different software is to have the software available on a website, where the creation of models, their visualization and manipulation are done online. In this way, any changes to the software can be checked against any effects on the model output. One of the first of these was JWS Online (http://jjj.biochem.sun.ac.za/), developed as a collaboration between the Vrije Universiteit Amsterdam in The Netherlands and the University of Stellenbosch in South Africa. WebCell is another such service for interactively exploring the steady-state and dynamic behaviors of models over the web. WebCell (http://webcell.kaist.ac.kr/) also incorporates analysis methods such as structural pathway analysis, metabolic control analysis and conservation analysis. A similar web service is available from the systems biology organization at the Keck Graduate Institute (http://sbw.kgi.edu/Simulation2005/).

"Web services and distributed computing are the only way to go," says Douglass Kell.

The National Resource for Cell Analysis and Modeling (NRCAM), located at the University of Connecticut Health Center and supported by the NIH, offers Virtual Cell (http://www.nrcam.uchc.edu/index.html), which uses a Java interface to allow the user to build models that are then automatically translated into the appropriate mathematical representation of ordinary or partial differential equations. These are then solved on the server and generate appropriate software code to perform and analyze simulations, and the results can be analyzed online or downloaded.

Douglas Kell, Director of the Manchester Center for Integrative Systems Biology at the University of Manchester, would like to see the modeling community adopt the idea of web services. "Web services and distributed computing are the only way to go. We found that every piece of software lacked one thing or another, none of them had it all." The idea is to have a distributed collection of software tools and data sets, a set of rules to make sure they talk together and a program that lets the user decide how information flows through these components. This is presently working well for programs like Taverna in the bioinformatics community. "Let users bolt it together as they choose," says Kell.

In many ways today's software is opening up the power of computational approaches to the wider biology community. As modeling services improve, computational modeling approaches will make further inroads into the lab.

Steven D. Buckingham is an investigator scientist at the Medical Research Council's Functional Genetics Unit at the University of Oxford (steven.buckingham@dpag.ox.ac.uk).

SUPPLIERS GUIDE: COMPANIES OFFERING PRODUCTS FOR MODELING, DATABASES, DATA MINING AND ANALYSIS

Company	Web address
Accelrys	http://www.accelrys.com/
Agilent Technologies	http://www.chem.agilent.com/
Ariadne Genomics	http://ariadnegenomics.com/products/
BG Medicine	http://www.bg-medicine.com/
Bioanalytics Group	http://bioanalyticsgroup.com/
Biomax Informatics	http://www.biomax.com/BIOMAX.COM/products/bioxm.php
BioSieve	http://www.biosieve.com/
CambridgeSoft	http://www.cambridgesoft.com/software/BioDraw/
CLC bio	http://www.clcbio.com/
Entelos	http://www.entelos.com/
Genedata	http://www.genedata.com/products/phylosopher/
GeneGo	http://www.genego.com/
Genomatica	http://www.genomatica.com/
Genomatix	http://www.genomatix.de/products/BiblioSphere/BiblioSpherePE1.html
Genstruct	http://www.genstruct.com/
Invitrogen	http://www.invitrogen.com/
InforSense	http://www.inforsense.com/
Ingenuity Systems	http://www.ingenuity.com/products/pathways_analysis.html
Innetics	http://innetics.com/
Insightful	http://www.insightful.com/industry/pharm/discovery.asp#microarray
InSilico discovery	http://www.insilico-biotechnology.com/
Integrated Genomics	http://www.integratedgenomics.com/ergo.html
Jubilant Biosys	http://jubilantbiosys.com/
Lion Bioscience	http://www.biowisdom.com/solutions/srs/
MathWorks	http://www.mathworks.com/
Medicel Ltd	http://www.medicel.com/
Microsoft	http://www.microsoft.com/
Numerica technology	http://numericatech.com/
Ocimum Biosolutions	http://www.ocimumbio.com/
Oracle	http://www.oracle.com/database/product_editions.html
Physiomics	http://www.physiomics-plc.com/
Premier Biosoft	http://www.premierbiosoft.com/
Protein Lounge	http://www.proteinlounge.com/epath3d/
SoftBerry	http://www.softberry.com/
Spotfire	http://www.spotfire.com/
Stratagene	http://www.stratagene.com/
Teranode Corp	http://www.teranode.com/
Wolfram Research	http://www.wolfram.com/