National Simulation Resource

NSR

University of Washington

Circulatory Mass Transport and Exchange

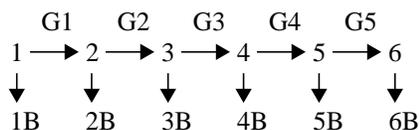## Application of multisubstrate modeling

I. Background

Two technical difficulties have interfered with the quantitative study of *in vivo* metabolism and biochemistry. One difficulty is acquiring high temporal resolution data describing cellular enzyme reactions. The other is accounting for the effects of capillary transport and exchange in the supply of reactants or removal of products. These difficulties are partially overcome by combining the multiple indicator dilution technique with multipathway, multiregion, axially distributed modeling.

The combined approach calls for new models that incorporate metabolic pathways into an established convection–diffusion model framework. One such model, based on a linear sequence of irreversible reactions, is named MSID4 (Multiple Substrate Indicator Dilution, 4 region).

II. Multisubstrate model information

The general model format and structure of MSID4 are identical to the single substrate model, MMID4. As in MMID4, twenty flow pathways model flow heterogeneity; large vessel operators model nonexchanging arterioles, venules and tubing; four regions model tissue structure and anatomy. Transport between regions is described by PS products, and metabolic transformation by G (consumption) terms. PSs and Gs are first order rate constants with identical units of clearance ($ml\ g^{-1}\ min^{-1}$).

The special features of the multisubstrate model (MSID4) involve its description of up to six tracer metabolites arranged in a linear irreversible series of reactions including irreversible sequestration:

$$
\begin{array}{cccccccccc}
& G1 && G2 && G3 && G4 && G5 \\
1 & \longrightarrow & 2 & \longrightarrow & 3 & \longrightarrow & 4 & \longrightarrow & 5 & \longrightarrow & 6 \\
\downarrow && \downarrow && \downarrow && \downarrow && \downarrow && \downarrow \\
1B && 2B && 3B && 4B && 5B && 6B
\end{array}
$$

The term G1 describes transformation of metabolites 1 to 2, so that the consumption of a parent metabolite results in the production of the daughter metabolite. Each metabolite is governed by its own independent set of transport and metabolic parameters. To describe irreversible sequestration, a variable fraction of each consumed tracer may be retained permanently in the model region in which it was formed. Here 1B, 2B, etc. describe irreversibly bound tracer.

The model was developed for describing the intravascular injection of a parent tracer compound, its entry into cells by the combined processes of flow and membrane influx, intracellular transformation to daughter metabolites, and the cellular and vascular efflux of unreacted parent and daughter metabolites produced in the tissue. The model is not limited to this experimental paradigm, because it can also provide solutions for tissue activity (residue functions) for the sum of all labeled metabolites, which is applicable to PET data analysis. The model can also be used to describe cellular production of metabolites without vascular inputs.

III. Multiple indicator dilution data

The MSID4 model was used for analyzing data on the sequence of three oxypurines: hypoxanthine → xanthine → uric acid. Both of the reactions are irreversible and are catalyzed by a single enzyme (xanthine oxidase/dehydrogenase). The xanthine oxidase/dehydrogenase reaction is of interest because it may produce oxygen free-radicals under some conditions, contributing to so-called "reperfusion injury", and it may provide the mechanism for macrophage cytotoxicity. Hypoxanthine is produced from the metabolism of adenosine and inosine. Data were obtained in multiple indicator dilution experiments in perfused guinea pig hearts, using an intracoronary bolus injection of [131-I]-albumin (intravascular reference), [3-H]-L-glucose (extracellular reference) and [14-C]-hypoxanthine. The fractionated venous samples were subject to high performance liquid chromatography and liquid scintillation counting analysis to measure outflow concentrations of the injected

Contents

Staff

Jim Bassingthwaighte, Director
jbb@nsr.bioeng.washington.edu

Joseph Chan, Programming
joseph@nsr

Harvey Friedman, Library
fnharvey@nsr

Rita Jensen, Administrative Asst.
jensen@nsr

Rick King, NSR Mgmt., SIMCON
rick@nsr

Keith Kroll, Modeling Applications
keith@nsr

Eric Lawson, Publications
eric@nsr

Gary Raymond, Applications
garyr@nsr

Larry Weissman, System Mgmt.
larryw@nsr

Figure 1. Model Fits to Dilution Curves of Hypoxanthine and Metabolites

Figure 1. Model Fits to Dilution Curves of Hypoxanthine and Metabolites

Legend:
- ■ Albumin
- ▲ L-Glucose
- ● Hypoxanthine
- □ Xanthine
- ○ Uric Acid
- — Model Fits
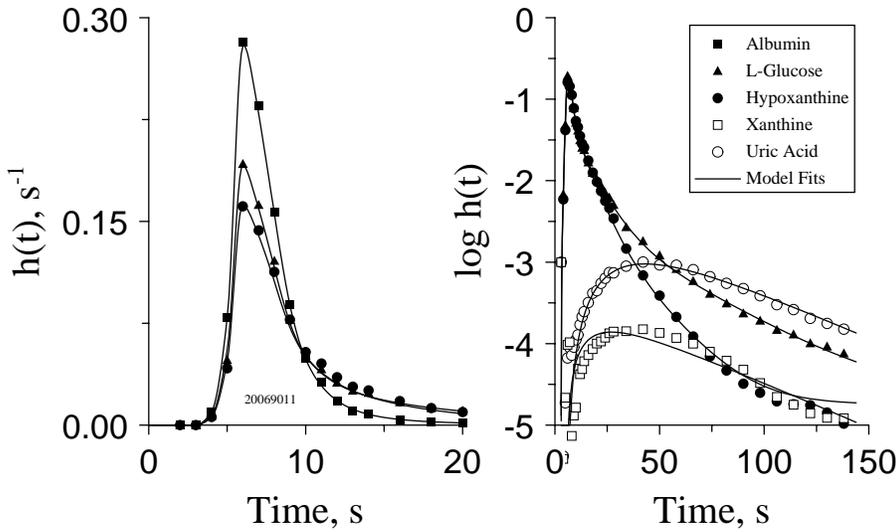
tracers and labeled xanthine and uric acid produced within endothelial cells. The curve peak is shown in figure 1, left panel (■, ●, ▲) on a linear scale of h(t), and the complete curves (▲, ●, □, ○) are shown in the right panel of figure 1 on a log scale of h(t) to display the metabolites more clearly. Xanthine and uric acid curves, emerging with a slower kinetic than precursor hypoxanthine, accounted for approximately 10% of injected hypoxanthine.
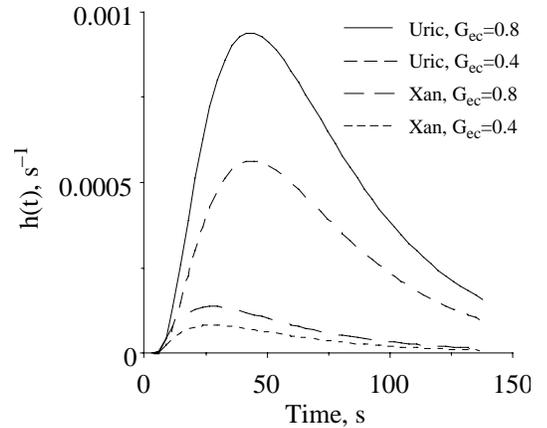
IV. Multisubstrate modeling

The structure of the tissue exchange unit is shown in figure 2. The xanthine oxidase/dehydrogenase pathway is restricted to the endothelial cell region, in accord with histochemical data. The purine salvage pathway, resulting in re-incorporation of hypoxanthine into the adenine nucleotide pool (irreversible sequestration), is described in the parenchymal cell region. Fol-

Figure 2. Structure of Model Exchange Unit

Figure 2. Structure of Model Exchange Unit
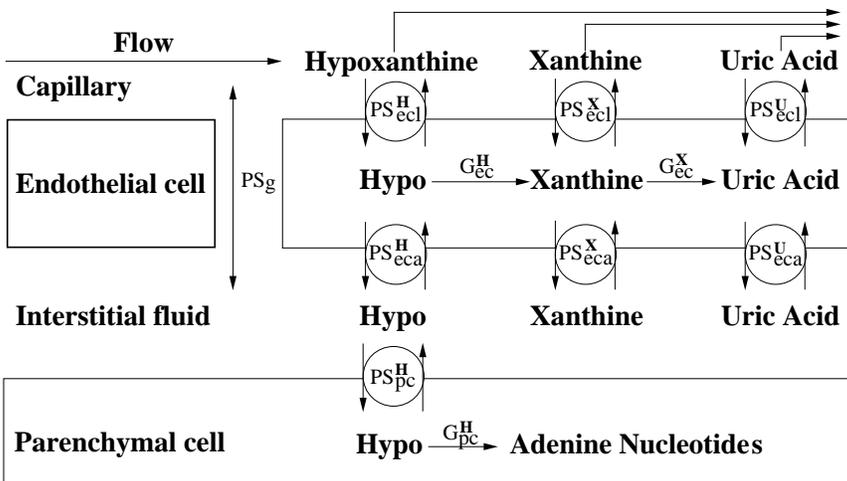
lowing standard modeling procedures, the albumin curve was used to obtain the model input function by deconvolution, and modeling the L-glucose curve was used to estimate extracellular transport parameters, such as PS_g and V_{isf}. The optimized model fits are shown by the continuous curves in figure 1. Because of the model structure, changes in parameters for parent metabolites affect the solutions for daughter metabolites. Therefore, fitting the curves of daughter metabolites places additional constraints on parameter estimates for the parent metabolites. The increased sensitivity is demonstrated in figure 3, which shows model solutions for uric acid and xanthine on a linear scale of h(t). The pair of solutions obtained with G_{ec} for hypoxanthine equal to 0.8 ml per g per min are identical to those in figure 1. Note the striking reduction in outflow concentrations in the other pair of solutions in which G_{ec} for hypoxanthine was decreased to 0.4 ml per g per min. Decreasing G_{ec} had a much smaller effect on the peak of the hypoxanthine curve itself.

Figure 3. Effect of G_{ec} for Hypoxanthine on Model Solutions for Uric Acid and Xanthine

Figure 3. Effect of G_{ec} for Hypoxanthine on Model Solutions for Uric Acid and Xanthine

Legend:
- —— Uric, G_{ec}=0.8
- – – Uric, G_{ec}=0.4
- – – Xan, G_{ec}=0.8
- - - Xan, G_{ec}=0.4

Although the indicator dilution technique is particularly well suited for studying endothelial cell metabolism, the multisubstrate model is also being used to analyze combined residue-outflow dilution data on the conversion of 15-O oxygen to 15-O water to assess regional myocardial oxygen consumption.

— Keith Kroll

Obtaining a vascular input function
via a stable, fast deconvolution technique

Deconvolution techniques have been studied elsewhere [1,2,3]. Deconvolution is necessary to estimate the input function when the output function of a linear stationary system and the system transfer function are known. Since the convolution process is commutative, one may use deconvolution to estimate the system transfer when the output function and input function to a linear stationary system are known.

Paynter proposed a method of numerical deconvolution [1] using Euler integration. The method is computationally inexpensive and fast, but the numerical solution may yield an unstable function, as mentioned by Paynter. The method proposed by Knopp, Dobbs, Greenleaf, and Bassingthwaighte [2] has had some success, but is not yet fully automated. Users decide what distribution, relative dispersion and skewness to use, and where to put the mean points of the distributions. The deconvolution with B splines proposed by Mendelsohn and Rice [3] has had some success too, but it is difficult to determine how many knots and what smoothness coefficient to use, as noted by Mendelsohn and Rice.

In MMID4, we use a hybrid regularization method which was developed here in our facility. The zeroth-order, first-order and second-order regularization methods are described by Beck, Blackwell and Clair [4]. Hybrid regularization is a judicious combination of both the matrix reduction technique and the regularization technique that yields an algorithm seemingly superior to either for this class of problem. We use any combination of zeroth-order, first-order and second-order regularization methods with the scalar chosen to be in proportion to the spectral radium of the lower triangular matrix consisting of the transfer functions.

The computational complexity of the regularization method increases quadratically as the number of points gets larger. Our output functions and the system transfer functions are unimodal and right skewed. The output functions derived from the experimental data are usually noisy. The system transfer functions calculated by model solutions are quite smooth.

Our goal is to obtain a smooth input function which may have a large number of points. First, we smooth the output function by a sum of exponential functions. Second, we deconvolute to get the front end of the input function containing the peak of the curve with a fine mesh. Third, we deconvolute the entire curve with a coarse mesh. Last, we integrate the results from the fine and coarse meshes to obtain the input function. The approach works well and is very efficient with our outflow dilution curves.

References

1. Paynter, H. M. Methods and Results from M. I. T. Studies in Unsteady Flow. *Boston Soc. Civil Eng. J.* 39: 2, (1952).

2. Knopp, J. T., Dobbs, W. A., Greenleaf, F. J., and Bassingthwaighte, J. B. Transcoronary intravascular transport functions obtained via a stable deconvolution technique. *Ann. Biomed. Eng.* 4:44-59, (1976).

3. Mendelsohn, J. and Rice, J. Deconvolution of Microfluorometric Histograms with B Splines, *JASA* 77:748-753, (1982).

4. Beck, J.V., Blackwell, B. and Clair, C. Jr. *Inverse Heat Conduction: Ill-posed Problem.* Wiley Interscience, 1985, New York.

— *Joseph Chan*

MMID4 update

Minor changes in MMID4 affect deconvolution and flow heterogeneity.

If nothing changes the deconvolution result between runs a small routine now bypasses deconvolution recalculation. Formerly, the user manually set a code to prevent recalculation. Recalculation is now prevented automatically. A similar routine reuses the flow heterogeneity calculation if nothing has affected its result.

An effort now underway to revamp the flow heterogeneity code will make it easier for a user to avoid having many marginal flow paths (paths with virtually no mass). This change is expected to be implemented in a short time.

SIMCON implementations

Since the time our last newsletter was published SIMCON has been implemented on two new platforms: Silicon Graphics, and the Aviion (Data General's UNIX workstation).

and installations

New installations include
— Seattle Veterans Administration Hospital
— University of Washington Dept. of Radiology
— Vanderbilt University (two installations)
— University of Illinois
— University of Southern California.

SIMCON update

The current version of SIMCON is 2.7.1. This version has nine modifications and three bug fixes that were not in the version extant when our previous newsletter was published.

Modifications

— Dependent variables can now be plotted versus another dependent variable as well as against the independent variable.

— The difference between a simulation output curve and a reference data curve (i.e., the residuals) can now be plotted.

— Curves from the input section of the reference data can now be plotted and can be used as the data curve for optimization. This necessitated a change in the format for the reference data curve codes, p(121-124), and optimization reference data curve index, p(196).

— A new menu, reference data index, has been added. This permits the user to examine the reference data that have been loaded and are available for analysis.

— SIMCON now has a new control parameter, the experimental run number. This controls the run from which data curves are selected for plotting, residuals, and optimization.

— The name and version of the model used to create output files and parameter database files is contained in the files. Whenever the files are read the user is advised if the current model name and version do not match.

— Parameter database files also contain the database version number. The user is asked whether the file should be updated if the current version number does not match. If the response is yes, all changes are made to bring the file up to date. This includes the changes in p(121-124) and p(196) mentioned above.

— Statistics are calculated automatically for each simulation output curve plotted. These statistics include the area, mean, relative dispersion, skewness, and kurtosis.

— New utilities for SIMCON programmers have been provided. These utilities are summarized on the manual page entitled "scputl".

Bugs

— The edit command now functions correctly when a parameter range is specified.

— When eight output curves are plotted, the tabular results file is no longer missing characters.

— Help is now accessible from all menus.

Future plans

Version 2.8.0 is now under development. This version will have two major enhancements: a second optimizer, SIMPLEX, will be available; a function generator, CINPUT, will be added. Except for bug fixes, this version will be the last version in the 2.N.N series. We will then begin moving version 3.0.0 from a prototype to a working version. This version will separate the interface and the model into two separate tasks and will make it possible to maintain the interface and models separately.

— Rick King

P.S.: Ever wonder what those version number mean? If the only changes are bug fixes, the third number is incremented (e.g., 2.7.0 → 2.7.1). If new features are added, the middle number is incremented. When major changes are made to the structure, such as separation into interface and model tasks, then the first number is incremented (e.g., 2.7.1 → 3.0.0).

Wanderings of bombadil

Note: bombadil is really a Sparcstation SLC which continues to help Prof. Deussen with simulation research in Duesseldorf. — *Ed.*

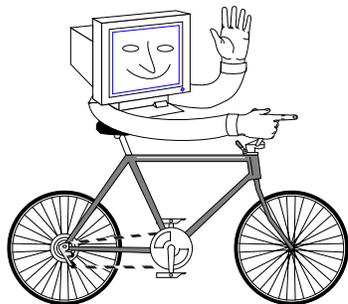Bombadil, who somewhere in his hypothalamic system was converted to frodo, is doing well.

He arrived on the last day of the year at the airport in Duesseldorf and was arrested by German customs. When I received the message of his arrest incognito from a nice fellow working for a transportation service, I drove to the airport immediately to start negotiations for his liberation. Persuasion took two long days before customs allowed that he could leave the bin and I obtained permission to transfer him to his new exile in the Heinrich-Heine University.

It was a dark Monday evening when I loaded bombadil onto the back seat of my car and drove southward through the center of Duesseldorf to the campus. Bombadil behaved like a real gentleman: he promptly agreed about his new refuge and booted right away. Despite the strenuous travel he hadn't lost any memory and started to learn about new things early the next morning.

Bombadil's environment is going to change over the next weeks. So far he is a stand alone fellow, but he will be connected to the network soon. His heart beats strong to resume communication with his earlier clan, all of their names being still clear in his front brain. Furthermore, his editing facilities will improve as soon as the desktop laser printer arrives.

— Andi Deussen

Vascular operators: modeling the larger vessels

A nonexchanging vessel is an artery, arteriole, venule, vein, or tubing segment. When an injection of a tracer moves through a nonexchanging vessel it can be delayed and dispersed. The duration of the delay, i.e., mean transit time, depends on the flow rate of the tracer through the nonexchanging vessel and the volume of the vessel; the formula for delay is V/F, where V is volume and F flow rate. Relative dispersion (RD) measures the spread of a signal around its mean; formally, RD is the standard deviation of an operator divided by its transit time.

Our computer code that simulates a nonexchanging vessel is called a 'vascular operator'. Vascular operators are described by their mean transit time and RD. To make a vascular operator, two simpler operators are used in series: a delay operator, and a fourth order operator (modeled by two slightly underdamped second order differential operators). Tracer flows through the delay operator and the fourth order operator sequentially.

Apportioning the volume of the vascular operator between the delay operator and the fourth order operator allows the vascular operator to have a variable RD. The RD of the delay operator is 0.0. The RD of the fourth order operator is fixed at 0.48. The RD of the vascular operator can be anything between 0.0 and 0.48.

The figure at right shows the response of a vascular operator to a sharp input function. As the RD increases, the height of the curve decreases, and it becomes skewed to the right. The peak of the curve shifts to an earlier time as more mass is shifted to later times.

In MMID4, these vascular operators are widely used to describe the circulation. For each tracer (intravascular, extracellular, and permeant) there are 44 vascular operators: 1 for the input tubing, 1 for the artery, 20 for arterioles, 20 for venules, 1 for the vein, and 1 for the output tubing. Any vascular operator may be eliminated from the system by setting its volume to 0.0. This allows the user to configure the MMID4 model in many different ways by eliminating or including vascular operators.

— *Gary Raymond*
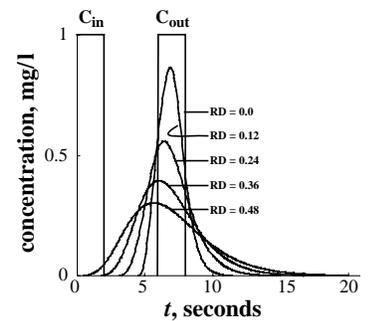
XSIM news

In the previous issue of the newsletter we announced that work had just begun on XSIM, the first member of a new generation of simulation interface programs. XSIM is based on the popular X11 window system that runs on nearly all UNIX workstations. Future members of this generation are planned for the Macintosh window system (MSIM) and for Microsoft's Windows environment (WINSIM).

Considerable progress toward implementing the major software components of XSIM has been made, but much work remains to integrate these components into a single user-friendly simulation interface. The components are

1. An extended facility for displaying model output graphically.

2. A static user interface for the model-independent or "SIMCON" portion of the display.

3. A dynamic display of hierarchical groups of variables associated with either the SIMCON or the model-specific portion of the program.

4. An interface language and fast parser for expressing how variables can be manipulated and displayed.

5. An interprocess communication (IPC) module that handles data transfer between the simulation interface and the current model.

The first component is implemented as a library of dynamic graphical objects. A prototype library was completed in early 1992, although Resource members continue to develop new useful objects. This library was also described in the March, 1992 issue of the newsletter. Because there is a limited amount of screen real estate available, graphics and controls will be placed in separate XSIM application windows. The user may elect to stack the windows, or may adjust their size to allow both to be displayed simultaneously. Windows can be separately resized and, if desired, iconified.

The second component, the static portion of the XSIM interface, is shown in the figure below. A table of buttons and status areas appears in the upper left, a scrollable diagnostic window in the upper right, and a large work space, where variables or named groups of variables will be displayed, in the lower portion of the window. When this screen snapshot was taken, the user had clicked the database button,
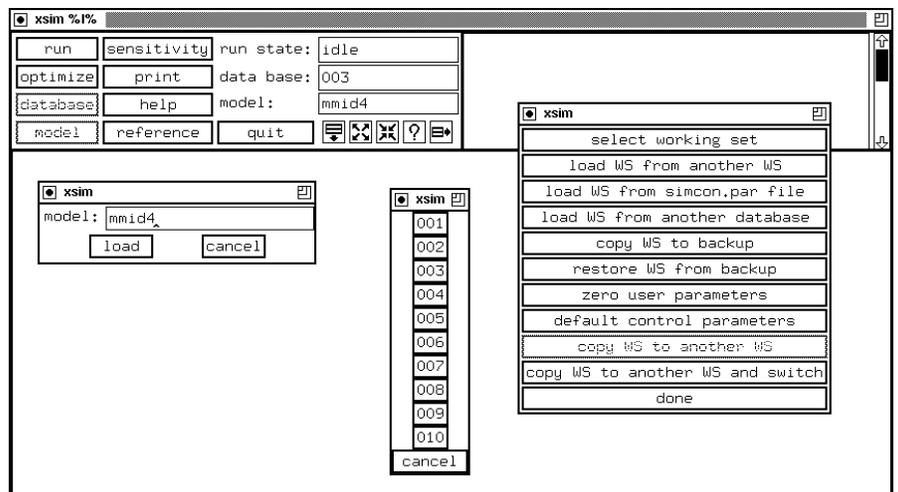
bringing up a pop-up menu, then clicked on "copy WS to another WS and switch" to bring up a list of working sets. The interface is waiting for a working set name (001 through 010) to complete the selection. The user also clicked on the "model" button, bringing up the model selection pop-up window. If the current model is changed, the old model (MMID4) will be released and a new one will be attached.

The next two components, the dynamic portion of the display and the fast configuration parser, are closely linked. The parser creates the data structures required for manipulating program variables and for displaying variables in the dynamic portion of the window.

The parser is designed to read plain text configuration files that describe variables and the display of variables. The configuration files, one for the model-independent control variables and one or more for each model, also specify preferences about variable grouping, initialization, and the style of display used for each variable. Some of the display details are described in the companion article "The flavor of XSIM" (next), and are shown in the figure accompanying that article.

The parser was built using the standard (but arcane) UNIX tools "lex" and "yacc". Free versions of these tools are also available under DOS and MacOS. With these tools, the configuration file syntax is easily changed to accommodate new features as the need arises.

At this time the parser, the data structure generator, and the X11-specific window display components are running. All are in the process of being integrated into XSIM.

The final component, the interprocess communications module, will be integrated last. This is one of several new features that allow XSIM to accommodate models written in languages other than Fortran. A prototype version works with the current SIMCON and could be grafted onto XSIM, but some improvements in the module are still being made.

XSIM is being built entirely with public domain software. It currently runs under a generic X11 Rev. 4 or Rev. 5 system. The extant version of XSIM uses a widget kit called "Xw", developed jointly by Digital and Hewlett-Packard Corporations. Xw is available in source form from the contributed portion of the MIT X11 distribution. Although XSIM is ICCCM-compliant, we expect that it will run on non-compliant X11 window managers. ☞

— Larry Weissman

The flavor of XSIM

XSIM supports "data objects" that consist of all the basic Fortran and C language data types, with homogeneous arrays under development. The only data type supported by SIMCON, and hence all current models, is Fortran REALs. This causes some unnatural data packing strategies for storing non-REAL (unreal?) data. Furthermore, XSIM data objects contain attributes in addition to their current value: for example, the upper and lower bound of numeric variables is stored, and whether the value and its limits can be modified by the user. This necessitates different display strategies that depend on the type of data object and preferences of the model writer.

The figure at left shows some of the variable types and typical display formats. In the sample figure, a variable named "Real" is limited to the range 0 and 100. The value can be changed by directly editing the current value field or by moving the slide bar. Entering a value outside these limits is rebuffed, and the value is set to the nearest limit. The sample box shows that the upper limit (max) is in an editable field, possibly with an ultimate upper limit set by the configuration file, but the lower limit (min) is a static label and therefore invariant. Another feature of the "Real" display is a square toggle button that indicates whether the variable is to be included in an optimization run. Clicking once on the title bar pops up a complete display of the variable's state. Selecting the title bar with an alternate mouse button pops up any associated help text.

The data structure for "Real" is generated by a configuration file definition that looks like the text in the box at top right, facing page.

The parameter definition gives the properties of the variable "Real" that are independent of its display. The group definition describes the parameters and any other objects that are displayed together. This allows a parameter to appear with more than one group of parameters, or even in no groups at all.

Several of the fields in each definition could have been defaulted, since there are global defaults and property inheritance rules for these items. For example, the keywords "opt" and "noopt" that indicate whether Real can be optimized has been omitted, since it defaults to "opt".

The format of the configuration file is C-like, and can be as terse or verbose as the model writer wishes. A model can even have more than one configuration file. This would be useful for
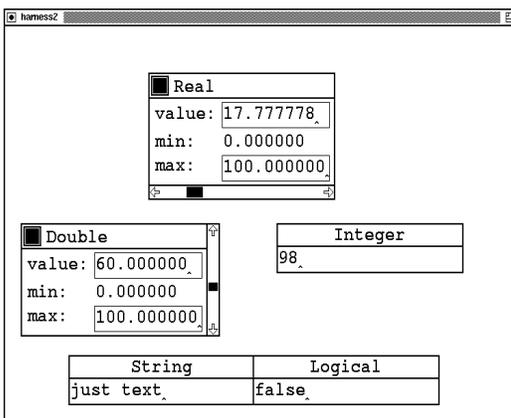
running a general model with a partial set of variables visible to allow a special case of analysis. For example, an abbreviated configuration file that uses a subset of capabilities of the more general MSID4 model may contain the definition appearing in the bottom box at right.

Given appropriate configuration files, all current models should be able to run under XSIM. Furthermore, the same models and configuration files should work as they stand on the PC and Macintosh versions of XSIM.

— *Larry Weissman*

```
parameter {
        name=demo_real,          ! internal handle
        wname="Real",            ! name in the title bar
        real,                    ! data type is real
        absmin=  0,  staticmin,  ! cannot be changed by user
        absmax=200,              ! can be changed
        loc=   104               ! P-array location (f77 models)
    }
...                              ! other parameters
group   {
        name=demo,
        par=(demo_real,  450, 800, sliderbelow),
        par=(demo_double, 50, 400, sliderright),
        ...
    }
```

```
model {
        name  = "msid4",                 ! model to run
        wname = "Oxygen-Water Exchange", ! name in the title bar
        type  = f77,                     ! how to talk to msid4
        ...                              ! other global defaults
    }
```

Selected NSR Publications

Bassingthwaighte, J. B., C. Y. Wang, and I. S. Chan. Blood-tissue exchange via transport and transformation by endothelial cells. *Circ. Res.* 65:997-1020, 1989.

Bassingthwaighte, J. B., I. S. Chan, and C. Y. Wang. *Blood-tissue exchange models: BTEX30 and BTEX40(UW/BIOENG-89/1). Report PB90-501396: FORTRAN Code; PB90-172578: Descriptive text* National Technical Information Services: Springfield, VA 22161, 1989.

Bassingthwaighte, J. B. Editorial. Interstitial adenosine: the measurement, the interpretation. *J. Mol. Cell. Cardiol.* 24:337-346, 1992.

Bassingthwaighte, J. B., R. B. King, and K. Kroll. A multisubstrate blood-tissue exchange model for the analysis of transients in cardiac tracer content. In: *Proceedings of the International Seminar on Imaging in Transport Processes, May 25-29*: International Centre for Heat and Mass Transfer, 1992, pp. 401-412.

Bassingthwaighte, J. B., A. Deussen, R. P. Beyer, I. S. Chan, G. R. Raymond, R. B. King, T. R. Bukowski, J. D. Ploger, K. Kroll, J. Revenaugh, J. M. Link, and K. A. Krohn. Oxygen transport in the myocardium. In: *Advances in Biological Heat and Mass Transfer HTD-Vol. 231*, edited by J. J. McGrath. New York: ASME, 1992, pp. 113-120.

Bassingthwaighte, J. B., I. S. Chan, and C. Y. Wang. Computationally efficient algorithms for capillary convection-permeation-diffusion models for blood-tissue exchange. *Ann. Biomed. Eng.* 20:687-725, 1992.

Bukowski, T., T. C. Moffett, J. H. Revkin, J. D. Ploger, and J. B. Bassingthwaighte. Triple-label β liquid scintillation counting. *Anal. Biochem.*:171-180, 1992.

Deussen, A., T. R. Bukowski, J. Ploger, J. M. Link, K. A. Krohn, and J. B. Bassingthwaighte. Three region, axially distributed model for the residue analysis of blood-tissue oxygen exchange and metabolism. *Pflügers Archiv. Eur. J. Physiol.* 420(Suppl.1):R131, 1992.

Gonzalez, F., and J. B. Bassingthwaighte. Heterogeneities in regional volumes of distribution and flows in the rabbit heart. *Am. J. Physiol.(Heart Circ. Physiol. 27)* 258:H1012-H1024, 1990.

Gorman, M. W., R. D. Wangler, J. B. Bassingthwaighte, D. E. Mohrman, C. Y. Wang, and H. V. Sparks. Interstitial adenosine concentration during norepinephrine infusion in isolated guinea pig hearts. *Am. J. Physiol. (Heart Circ. Physiol. 30)* 261:H901-909, 1991.

Kroll, K., A. Deussen, and I. R. Sweet. Comprehensive model of transport and metabolism of adenosine and S-adenosylhomocysteine in the guinea pig heart. *Circ. Res.* 71:590-604, 1992.

Kroll, K., T. R. Bukowski, L. M. Schwartz, D. Knoepfler, and J. B. Bassingthwaighte. Capillary endothelial transport of uric acid in the guinea pig heart. *Am. J. Physiol.(Heart.Circ.Physiol.31)* 262:H420-H431, 1992.

Kuikka, J. T., J. B. Bassingthwaighte, M. M. Henrich, and L. E. Feinendegen. Mathematical modeling in nuclear medicine. *Eur. J. Nucl. Med.* 18:351-362, 1991.

Shields, A. F., M. M. Graham, S. M. Kozawa, L. B. Kozell, J. M. Link, E. R. Swenson, A. M. Spence, J. B. Bassingthwaighte, and K. A. Krohn. Contribution of labeled carbon dioxide to PET imaging of carbon-11-labeled compounds. *J. Nucl. Med.* 33:581-584, 1992.

Wong, A. Y. K., A. Fabiato, and J. B. Bassingthwaighte. Model of calcium-induced calcium release mechanism in cardiac cells. Bull. Math. Biol. 54:95-116, 1992.

JBB
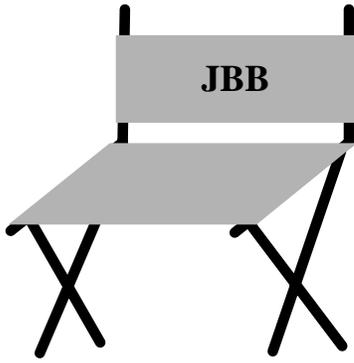
Director's Corner

Summer Course, 1992

From August 24-28 we gave our intensive (as usual) course "5th Annual Computer Simulation Summer Workshop on Circulatory Transport and Exchange Processes" and even included a little material on fractals and chaos. This was our best ever effort, for we had the advantage of being in a properly equipped computerized teaching room connected to the Resource network. Thus it was easier than in past years for the "students" to do the modeling right from the first morning of the week.

Next year's course (projected June 20–25), will put more emphasis on coding new models and nonlinear processes for transport and metabolism.

Years of grant writing and moments of a site visit

The last two years seemed to last almost forever. All of our grant programs came up for review, and the insecurity level and the writing speed rose to (or for, or in spite of) the occasion.

NSR submitted an application for renewal to the National Center for Research Resources in January, and our site was visited in July. The Division of Research Grants honored us by selecting a sterling group of reviewers expert in biology, medicine and computer science.

The favorable result, another five years of funding, supports the principle that research service and education can go together, and that our efforts, though improvable in many ways, were in the right direction.

Our equipment and system configuration will remain much as described in the previous Newsletter, because equipment requests for the third year of the renewal were not funded.

New projects include one on developmental biology of the lung, by James D. Murray, who is now at the University of Washington while retaining an appointment at Oxford. Complex biochemical systems modeling, on the other hand, was not supported: I believe this area to be of great long term importance, and so will look to our advisory committee to examine this issue in the their 1993 review.

— Jim Bassingthwaighte

61-0719
Eric Lawson, Publications
Bioengineering, WD-12
University of Washington
Seattle, WA 98195