SAAM II: Simulation, Analysis, and Modeling Software for Tracer and Pharmacokinetic Studies

P. Hugh R. Barrett, Bradley M. Bell, Claudio Cobelli, Hellmut Golde, Alan Schumitzky, Paolo Vicini, and David M. Foster

Kinetic analysis and integrated systems modeling have contributed substantially to our understanding of the physiology and pathophysiology of metabolic systems and the distribution and clearance of drugs in humans and animals. In recent years, many researchers have become aware of the usefulness of these techniques in the experimental design. With this has come the recognition that the discipline of kinetic analysis requires its own expertise. The expertise can impact experimental design in many ways, from the collaborative and service activities in which individuals interact in formal ways to the development of software tools to aid in kinetic analysis. The purpose of this report is to describe one such software tool, Simulation, Analysis, and Modeling Software II (SAAM II). In the first part, we describe in general how the user can take advantage of the capabilities of the software system, and in the second part, we give three specific examples using multicompartmental models found in lipoprotein (apolipoprotein B [apoB] kinetics) and diabetes (glucose minimal model) research.

KINETIC ANALYSIS and integrated systems modeling have contributed substantially to our understanding of the physiology and pathophysiology of metabolic systems in humans and animals. Although research articles may state which specific software system was used in the data analysis, these and most review articles in the biomedical literature contain little information on software packages available for tracer and pharmacokinetic analysis in general, or their application to specific areas such as lipid and lipoprotein metabolism, intermediary metabolism, or trace element and mineral metabolism. The purpose of this report is to describe one such package, the Simulation, Analysis, and Modeling Software II (SAAM II) system, and how it can be used in experimental design and data analysis.

The SAAM II system, a totally reengineered software system based on Berman's SAAM,[1] is a powerful research tool to aid in the design of experiments and the analysis of data. SAAM II deals easily with compartmental and numerical models, helping researchers create models, design and simulate experiments, and analyze data quickly, easily, and accurately. Compartmental models are constructed graphically using drag and drop icons. The models can be either linear or nonlinear. Numerical models are created by entering algebraic equations directly.

Computationally, SAAM II creates systems of ordinary differential equations from a specified compartmental model structure, and simulates solutions given specific parameter values and input information. The software fits models to data using a powerful new optimization technique, and provides statistical information about the fit.

We will first describe how models are constructed and solved using SAAM II. We will then discuss three specific examples that take advantage of some of the advanced modeling features of SAAM II.

From the Resource Facility for Kinetic Analysis, Department of Bioengineering, and Applied Physics Laboratory, University of Washington, Seattle, WA; Department of Electronics and Informatics, University of Padova, Padova, Italy; and Department of Mathematics, University of Southern California, Los Angeles, CA.

Submitted October 3, 1997; accepted October 3, 1997.

Address reprint requests to David M. Foster, PhD, Resource Facility for Kinetic Analysis, Department of Bioengineering, Box 352255, University of Washington, Seattle, WA 98195-2255.

0026-0495/98/4704-0021$03.00/0

THE COMPARTMENTAL MODULE

In the compartmental module, the user can choose from a set of drag and drop model-building icons representing compartments, transfers, and delays to build a visual representation of a system on a drawing canvas. For each icon, attributes can be defined using dialogue boxes. For example, reference names can be assigned to compartments.

To illustrate the model-building capability, consider the commonly used Matthews model.[2] This is a two-compartment model with a central (plasma) compartment, from which irreversible loss occurs, and an exchange (peripheral) compartment.

To simulate the behavior of the model, an experiment must be created. Using the drag and drop experiment-building icons, the user directly specifies exogenous inputs and sample sites. Each has an attribute box where, for example, measurement equations are written and associations with data elements are made. Figure 1 shows an example of an experiment specified on the Matthews model.

There are a number of important features associated with the experiment-building icons. The attribute box associated with the input allows the user to specify any functional form for the input function, for example, a bolus injection, a continuous infusion, a primed-continuous infusion, or a generic equation. The attribute box associated with the sample is where one writes the measurement equation. This is the equation that translates the units of the differential equations (usually compartmental masses) to the units of the actual data (usually concentrations or tracer to tracee ratios). In this dialogue box, the user also associates the sample with a specific data set. The data are entered separately in a data table that can be created internally or imported from a spreadsheet. Finally, parameter values are entered using the parameter-entry dialogue box.

A compartmental model is in fact a graphical formulation of a system of ordinary differential equations.[3] Once the user specifies the structure and the inputs and outputs, SAAM II will automatically create the corresponding system of ordinary differential equations from the model structure. Figure 2 shows the equations defined internally by SAAM II, together with a set

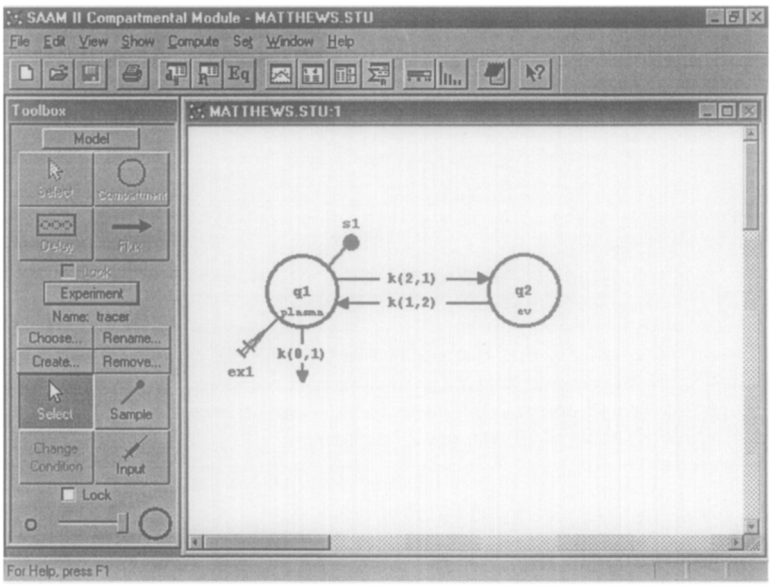

Fig 1. Constructing the Matthews model using SAAM II. The compartment model-building icon is used to create compartments 1 and 2; an attribute box associated with the compartments is used to enter the reference names "plasma" and "ev." Transfers between, into, and leaving the compartments are constructed using the flux model-building icon. The drag and drop icons used to generate the input and sample, ie, the syringe and solid line with the bullet, are indicated on the canvas.

of user-defined equations. In Figure 2, the last two equations listed in the read only part of the box are the actual differential equations that SAAM II integrates numerically (see COMPUTATIONAL ALGORITHMS).

The model output can be displayed in graphical or tabular form, printed, and saved to a file. Figure 3 is an example of the statistical information obtained following a successful fit of the Matthews model (Fig 1) to a set of low-density lipoprotein (LDL) apolipoprotein B (apoB) tracer data obtained following a bolus injection of radioiodinated LDL.

THE NUMERICAL MODULE

In the SAAM II numerical module, the user can specify the model directly by entering algebraic equations, or choose from a collection of predefined models that include polynomials, sums of exponentials, sums of Gaussians, Michaelis-Menten equations, and Scatchard analyses.

A special feature of the sums of exponentials option makes estimating noncompartmental parameters easy.[4] As with the compartmental module, SAAM II solves the (algebraic) equations and fits them to the data. The output is the same as that described for the compartmental module.

Figure 4 describes an example of fitting a sum of exponentials to a set of data following an infusion of tracer-labeled material where a washout period is included in the experimental design.

In this example, two functions are required: one to describe the increasing portion of the data during the constant infusion, and the other to describe the washout phase. The former is given by yar (t) and the latter by yaw (t) in Fig 4. The function that is actually fitted to the data, y(t), is written

$$y(t) = \text{if } (t < 300) \text{ then yar(t) else yaw(t)} \qquad \text{Eq 1}$$

This means that when the time is less than 300, the time at

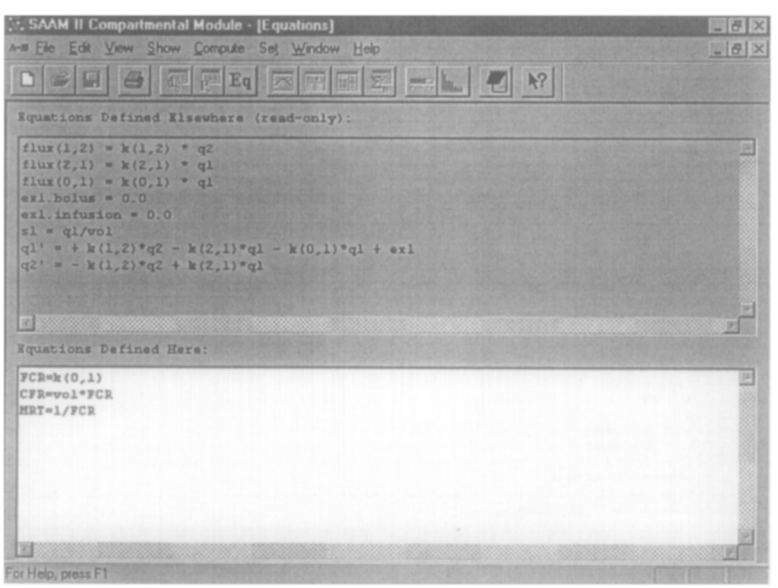

Fig 2. The equations that are created internally by SAAM II are displayed in the "read-only" portion of the equations dialog box. The user has the option of entering auxiliary equations in the lower part of this box. Here, for example, equations for the fractional catabolic rate (FCR), the fractional clearance rate (CRF), and mean residence time in plasma (MRT) are written.

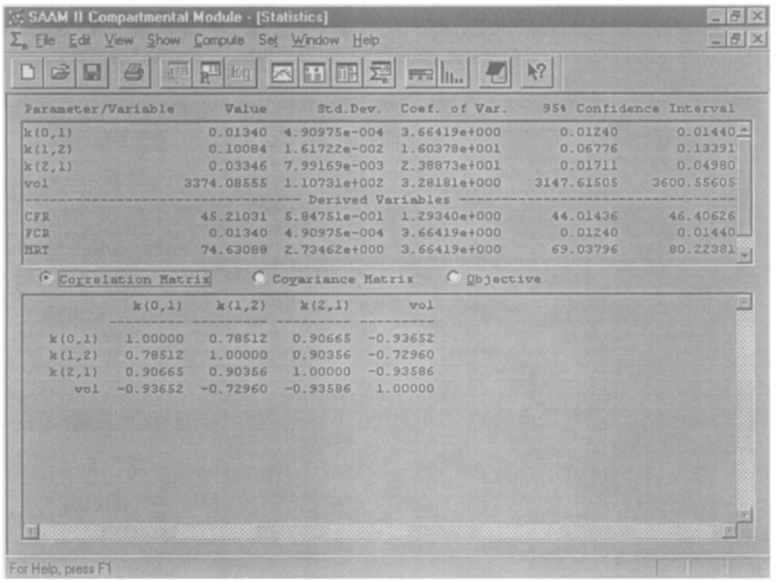

Fig 3. The statistical information available following a successful fit of the Matthews model to a set of data. Best estimates for both the primary parameters and derived parameters (functions of the primary parameters), their estimated standard deviation, their fractional standard deviations, and 95% confidence limits are provided. The correlation coefficients are also provided. Additional options include the covariance matrix and information on the objective function. This information can be written to a file for post-processing by standard statistical packages.

which the infusion stops, the data are described by yar (t); otherwise, they are described by yaw (t).

A plot of the output following a fit of this model to a set of data is shown in Fig 5.

COMPUTATIONAL ALGORITHMS

There are two basic numerical computations performed by SAAM II, solving systems of ordinary (linear or nonlinear) differential equations and optimization. The program automatically constructs differential equations from the model that is built on the screen. Parameters within the model that are not specified by an equation can be estimated by the optimization procedure.

SAAM II provides a choice of three integration methods. The Rosenbrock integrator uses a semi-implicit method. The Runge-Kutta integrator uses a standard forward-integrating order 5-4 method. The Pade integrator uses a method developed by the Resource Facility for Kinetic Analysis at the University of Washington; it is based on the Pade approximation of the matrix exponential. Each integrator has different strengths and weaknesses, which will not influence the correctness of the results but may dramatically influence the running time of a solve or fit operation. The default integrator is the Rosenbrock integrator, because it solves the largest class of models. The Pade integrator is applicable only when the model has constant transfer rates and bolus and constant-infusion inputs. The Rosenbrock integrator is appropriate for stiff models, and the Runge-Kutta integrator is most useful for nonstiff models.

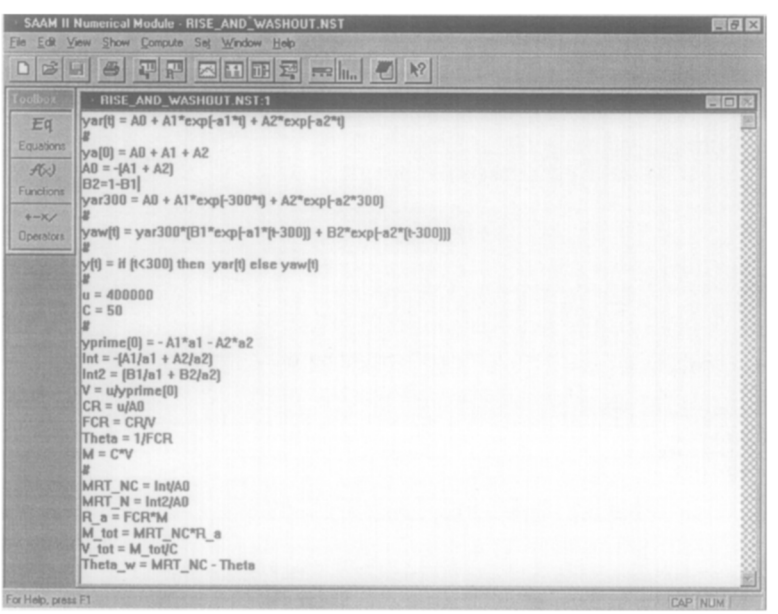

Fig 4. An example of using the numerical module to estimate the noncompartmental parameters in a protocol where tracer was given as a constant infusion followed by a washout period. See text for additional information.

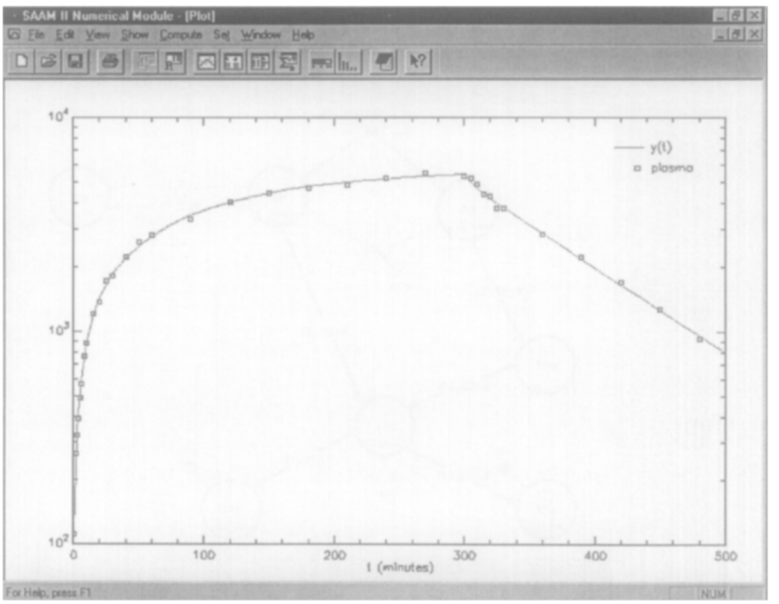

Fig 5. A best fit of the function y(t) described in Eq 1 to a set of data.

The optimizer is a modification of the Gauss-Newton method that handles the case of multiple datasets, where data sets can differ numerically by several orders of magnitude. The theory for this estimator and the algorithm have been described in detail.[5] Briefly, in the general case of J data sets, each one containing N_j, $j = 1, \ldots, J$, data points, the objective function in SAAM II is the extended least-squares maximum likelihood function:

$$OBJ_{SAAM} = \frac{1}{M}\sum_{j=1}^{J}\sum_{i=1}^{N_j}\left\{\frac{[y_{i,j} - s(\hat{p}, t_{i,j})]^2}{V_{i,j}[s(\hat{p}, t_{i,j}),y_{i,j},\hat{v}_j]} + \log V_{i,j}[s(\hat{p}, t_{i,j}),y_{i,j},\hat{v}_j]\right\} \quad \text{Eq 2}$$

where

- M is the sum of all data points across all data sets: $M = N_1 + N_2 + \ldots + N_j$,
- $y_{i,j}$ and $t_{i,j}$ are the i-th datum and time point, respectively, in the j-th data set,
- $\hat{p}$ is the vector of estimated parameters,
- $s(\hat{p},t_{i,j})$ is the model prediction at time i in the j-th data set,
- $V_{i,j}[s(\hat{p},t_{i,j}),y_{i,j},v_j]$ is the measurement error variance of the i-th datum in the j-th data set. Note that it can be a function of the model prediction, $s(\hat{p},t_{i,j})$, or the data value, $y_{i,j}$. The generic form of the variance used in SAAM II is $V_{i,j} = A + By_{i,j}^c$ in the case of data-based weighting or $V_{i,j} = A + Bs(\hat{p},t_{i,j})^c$ in the case of model-based weighting (A, B, and C are nonnegative constants; note that for $B = 0$ the error variance is constant and for $A = 0$ the coefficient of variation of the error is constant). The weight of the datum is therefore the inverse of the variance of the datum at that time (as is theoretically correct),
- $\hat{v}_j$ is the a posteriori variance factor of the j-th data set. This is relevant when the measurement error variance is known up to a proportionality constant: this is the case of relative weighting, where $\hat{v}_j$ is estimated from data, as opposed to absolute weighting, when the measurement error variance is exactly known (and $\hat{v}_j$ is then assumed equal to 1).

This formulation of the objective function allows adjustment of the parameters not only to achieve an optimal fit to the data but also (in the case of model, relative weighting) to optimize the variance of the data with respect to the available information. Asymptotic parameter precisions are given by SAAM II for all available weighting schemes (data- or model-based and absolute or relative).

Using an extension of the objective function, SAAM II can also perform Bayesian fitting. This is useful, for example, when the mean and standard deviation of a model parameter within a given population are known. The availability of this prior information can then be incorporated into the fitting process for a single individual belonging to the same population, thus allowing estimation of model parameters with improved confidence.

EXAMPLES

A Multicompartmental Model for LDL ApoB Kinetics With Plasma Heterogeneity

A comprehensive model for LDL apoB kinetics was developed.[6] This model was derived from an experimental protocol wherein a bolus of radioiodinated LDL was injected, and serial plasma and individual urine samples were collected for 14 days. The integrated kinetic modeling revealed kinetic heterogeneity. In the report by Foster et al,[6] the most commonly used model was "model B." Figure 6 shows how this model would be constructed using SAAM II.

This model uses several special features in SAAM II. First, the plasma sample s1 is a sum of the radioactivity predicted in

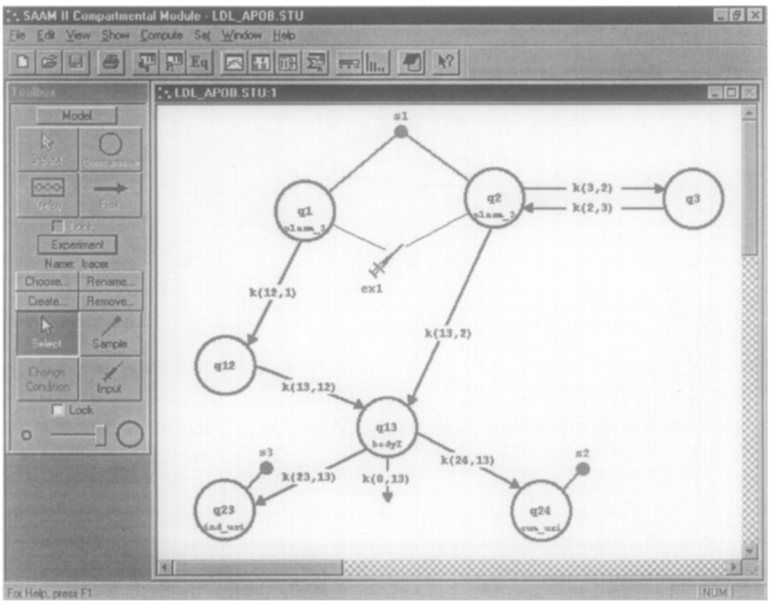

Fig 6. A multicompartmental model for LDL apoB kinetics. A bolus of radioiodinated material is injected into compartments q1 and q2, and the sample s1 is a sum of the material in q1 and q2. The sample is associated with plasma data. Compartment q2 exchanges with an extravascular compartment q3. The radioiodide resulting from the catabolism of the substance appears first in the body iodide pool, and is then collected in urine. Compartments q5 and q6 respectively represent the individual 24 hour and cumulative urine collections. Samples s2 and s3 are associated with these data.

compartments 1 and 2. The sample equation is

$$s1 = (q1 + q2)/vol, \qquad \text{Eq 3}$$

where q1 and q2 are the model-predicted radioactivity in compartments 1 and 2 and vol is plasma volume, a parameter estimated from the data. Next, the initial input is split between compartments 1 and 2; the percentage of the total injected amount in these compartments also is a parameter estimated from the data.

Following a successful fit of the model to a set of data, the model output can be viewed graphically. Figure 7 shows a plot of the data with the model-predicted time courses.

The use of individual 24-hour urine samples requires an additional explanation because it uses another advanced feature of the software. To simulate 24-hour urine collection, which in the model shown in Fig 6 is compartment 23, the numerical value in compartment 23 must be reset to 0 at 24-hour intervals. This is very easily accommodated in SAAM II, and is reflected in the plot shown in Fig 10. The plot is in fact sawtooth, reflecting the fact that the contents of compartment 23 are set equal to zero daily; thus, the user can see how the daily urine accumulates.

Using a Forcing Function to Describe Amino Acid Incorporation Into ApoB

In the past decade, there has been a move away from the use of radioactive isotopes to stable isotopes to trace the metabolism of lipoproteins. With this, a change has come in the way lipoproteins are labeled. Most radioactive-tracer studies exogenously labeled the apolipoproteins of interest. However, today,

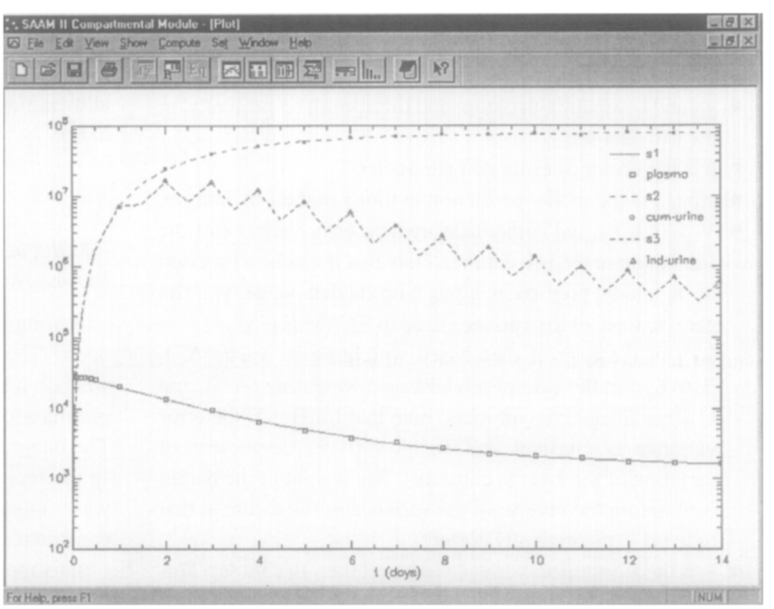

Fig 7. A semilogarithmic plot of the model-predicted values (lines) and data (symbols) following a successful fit of the model shown in Fig 6 to a set of data. Data shown are plasma concentration, individual 24-hour urine samples, and cumulative urine samples. See text for additional information.

apolipoproteins are labeled endogenously following administration of a labeled amino acid such as ^{13}C-leucine.

The kinetics of amino acids are complex,[7] so a possible approach to incorporating plasma amino acid data into the development and fitting of a compartmental model to tracer data is to use a forcing function, ie, to equate the mass of a given compartment to some previously measured time course. The use of a forcing function obviates the need for a complex model to describe the kinetics of the amino acid, because this mathematical description embodies the recycling processes of the tracer. In this way, the system can be decoupled and the plasma amino acid data can be used as the source of tracer. A three-exponential equation was used as the forcing function to drive the appearance of tracer into the apoB model shown in Fig 8 and described in the study by Parhofer et al.[8] In this study, some subjects received a primed constant infusion of an amino acid for a period of 8 hours, after which the infusion pump was turned off. The duration of the turnover studies was approximately 110 hours, during which time plasma samples were collected and amino acid tracer to tracee ratios were determined in plasma amino acid, VLDL apoB, IDL apoB, and LDL apoB fractions.

In SAAM II, a forcing function is defined by opening the attributes dialogue box of the compartment and defining it as a forcing function. A forcing function compartment is marked with an "FF," as in the Figure. A forcing function can be specified in two ways: by using data or by using an equation. In the former case, the compartmental mass is given by the time course of the specified data (linearly interpolated); in the latter, it is an algebraic equation (function of time or of any other parameter in the model) that defines it.

In an apoB turnover study where an amino acid is given as a primed infusion for a period of 8 hours and is then tracked during the washout phase, a three-exponential equation is required to describe the plasma amino acid data. For the purpose of this example, we have defined the forcing function, q1.FF, to equal the function Leu_FF_eqn. The equation is as follows:

$$\begin{aligned}
&\text{Leu_FF_eqn} = \text{swit} * \text{rise} + \text{fall}\\
&\text{rise} = A0 + A1*\exp(-a1*t) + A2*\exp(-a2*t)\\
&\qquad + A3*\exp(-a3*t)\\
&\text{rise8} = A0 + A1 * \exp(-a1*8) + A2*\exp(-a2*8)\\
&\qquad + A3*\exp(-a3*8)\\
&\text{fall} = (1\text{-swit}) * (B1 * \text{rise8} * \exp(-a1*(t-8))\\
&\qquad + (1\text{-}B1) * B2 * \text{rise8} * \exp(-a2*(t-8))\\
&\qquad + (1\text{-}B1) * (1\text{-}B2) * \text{rise8} * \exp(-a3*(t-T)))
\end{aligned} \qquad \text{Eq 4}$$

The equation is divided into two parts, one that describes the rise of the amino acid curve, called rise, and one that describes the fall of the curve, called fall. Because rise and fall have nonzero values throughout the entire duration of the experiment, it is necessary to use a switch variable to turn on the rise section of the curve during the period the primed infusion is on and then, at 8 hours, to turn off the rise equation. At 8 hours, the value of the switch variable changes instantaneously from 1.0 to 0.0. At the start of the experiment, the value of the switch is set to 1.0, meaning that the fall equation does not contribute yet to the amino acid forcing function curve. Then at 8 hours, when the switch variable changes from 1.0 to 0.0, the effect is to turn on the fall equation contribution to the forcing function by virtue of the way the equation for fall is written (note that it contains the (1-swit) term, which becomes equal to 1 after 8 hours). The value of the switch variable is changed using the change conditions dialogue window in SAAM II: this requires specifying a time at which the variable changes value. The falling component of the Leu_FF_eqn equation uses the value of the rising function at 8 hours as the starting point for the falling component.

With this equation, one can fit the plasma amino acid data and

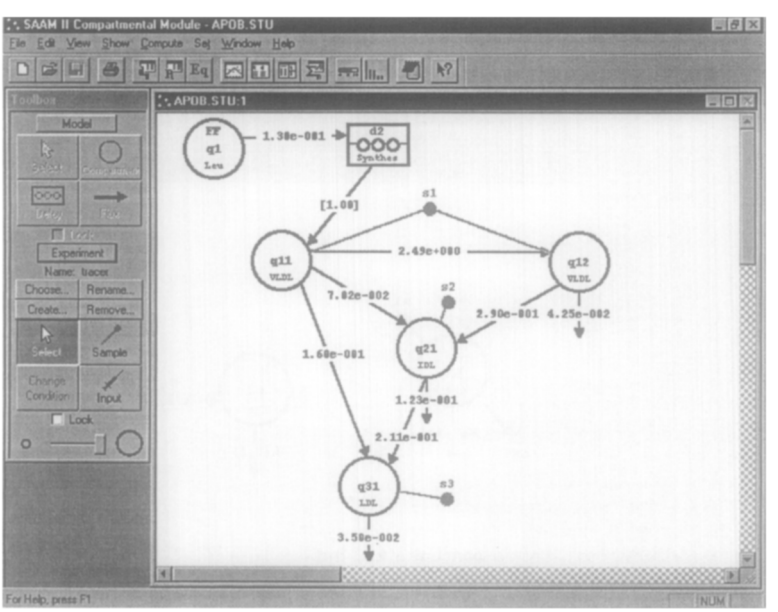

Fig 8. Compartmental model describing the kinetics of apoB in normolipidemic individuals.[8] Compartment q1 represents the plasma amino acid pool and is described using a three exponential forcing function. Compartment d2 is a delay compartment that is required to account for the time of synthesis and secretion of VLDL apoB particles. Compartments q11 and q12 represent plasma VLDL apoB, sample s1 is the sum of the two VLDL compartments and is associated with the VLDL apoB tracer data. Compartment q21 is plasma IDL apoB and sample s2 represents the IDL apoB tracer data. Compartment q31 is plasma LDL apoB, sample s3 is associated with the LDL apoB tracer data. The numbers on each transfer represents the fractional transfer rate (h^{-1}) for that pathway.

subsequently use the same equation as the forcing function to drive the appearance of labeled amino acid into VLDL apoB. The interested reader is referred to Parhofer et al[8] for details.

Glucose Minimal Models

A method to noninvasively estimate indices describing the control of glucose and insulin on glucose production and disposal in normal and disease states is of utmost importance. The classic[9,10] and the hot[11] minimal models of glucose kinetics have been proposed for this purpose; they are now widely used in physiological studies. The models are nonlinear compartmental models with parameters identified from standard (IVGTT) or tracer-labeled (HIVGTT) intravenous glucose tolerance test data, respectively.

The classic minimal model. The classic minimal model[10] is described by the two equations,

$$\frac{dG(t)}{dt} = -[S_G + x(t)]G(t) + S_G G_b \quad G(0) = G_0 \quad \text{and}$$
$$\frac{dx(t)}{dt} = -p_2[x(t) - S_I[I(t) + I_b]] \quad x(0) = 0, \qquad \text{Eq 5}$$

where G(t) and I(t) are glucose and insulin concentrations in plasma, respectively, G_b and I_b are basal glucose and insulin concentrations in plasma, respectively, G_0 is the nonzero intercept of the model-predicted glucose concentration, x(t) is insulin action quantified by p_2, and S_G and S_I are parameters of glucose effectiveness and insulin sensitivity, respectively. S_G measures the effect of glucose per se at basal insulin to stimulate glucose disposal and to inhibit endogenous production, while S_I measures the ability of insulin to enhance the glucose stimulation of glucose disposal and the glucose inhibition of endogenous glucose production. It must be appreciated that these parameters measure the effect of glucose and insulin, respectively, both to stimulate plasma glucose disappearance and to inhibit endogenous glucose release. The plasma insulin concentration (above basal) time course during the test, I(t), acts as an input to the model. From insulin and glucose data, the uniquely identifiable parameters are S_G, p_2, S_I, and G_0.

The hot minimal model. The hot IVGTT consists of adding to the IVGTT bolus a glucose radioactive or stable isotope and measuring in the plasma, in addition to insulin and glucose, the hot glucose concentration.[11] Addition of the tracer permits the segregation of glucose disappearance from endogenous glucose release processes. The idea is thus to perform minimal modeling of the insulin-hot glucose concentration data to extract peripheral tissue indices of glucose effectiveness and insulin sensitivity. The hot minimal model is described by the equations,

$$\frac{dG^*(t)}{dt} = -[S_G^* + x^*(t)]G^*(t) \quad G^*(0) = G_0^* \quad \text{and}$$
$$\frac{dx^*(t)}{dt} = -p_2^*[x^*(t) - S_I^*[I(t) + I_b]] \quad x^*(0) = 0, \qquad \text{Eq 6}$$

where the superscript "*" refers to parameters associated with hot glucose. The uniquely identifiable model parameters are S_G^*, p_2^*, S_I^*, and G_0^*. The model is identified from insulin and tracer glucose data, and provides new indices of glucose effectiveness, S_G^*, and insulin sensitivity, S_I^*. S_G^* and S_I^* reflect only glucose disposal processes: S_G^* measures the ability of glucose per se at basal insulin to stimulate glucose disposal, and S_I^* measures the ability of insulin to enhance glucose per se stimulation of glucose disposal.

Implementation of the minimal models using SAAM II. Both the cold and the hot single-compartment minimal models can be easily implemented in the SAAM II environment. The implementation uses the forcing function feature described in the previous example; here, the forcing function is used to describe plasma insulin levels.

The cold minimal model has as inputs the initial condition, G_0, and the constant term, $S_G G_b$. These correspond to initial condition $G(0) = G_0$ and the term $+S_G G_b$ in the differential equation dG/dt. The measured output is the glucose concentration in plasma, G(t). The forcing function that describes plasma

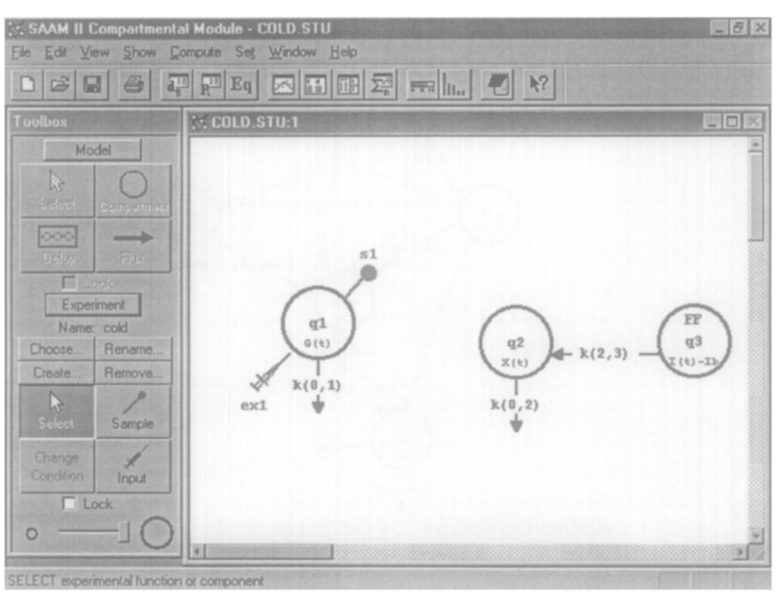

Fig 9. The classical glucose minimal model. Compartment 1 is plasma glucose, *G(t)*. Compartment 2 describes insulin action *x (t)*. See text for additional information.

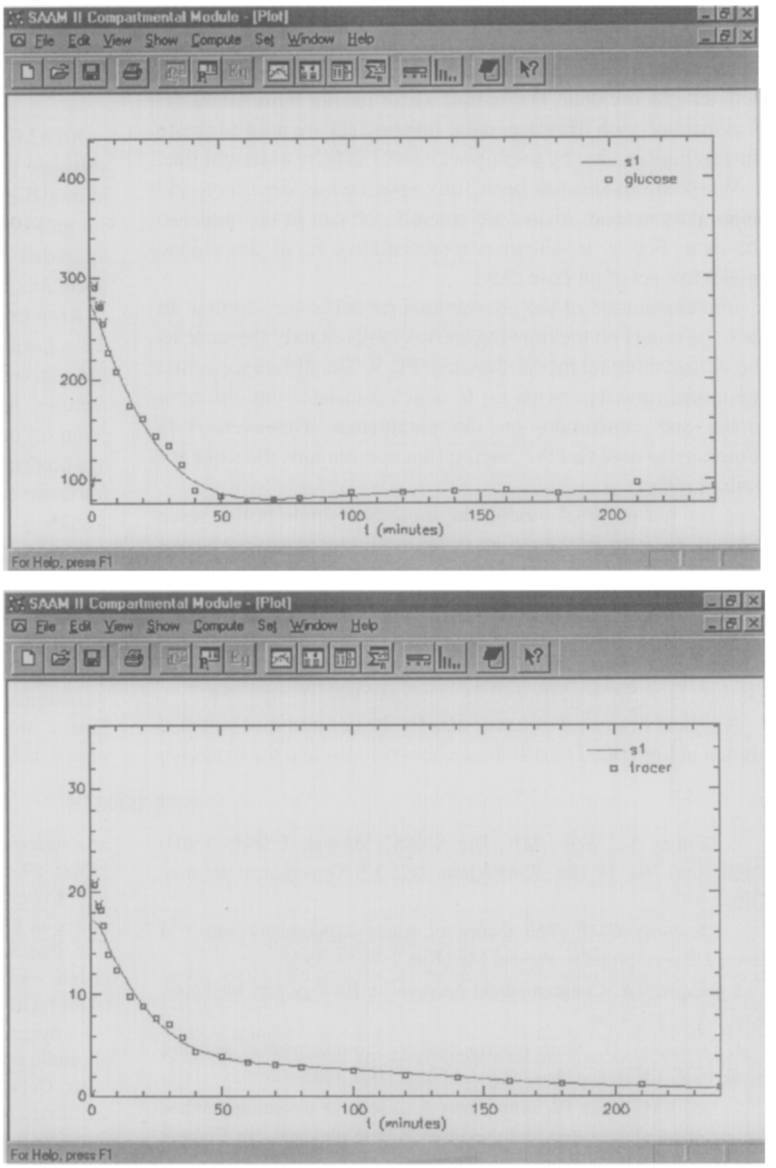

Fig 10. (Top) A plot of the best fit of the classical glucose minimal model to a set of plasma glucose data. (Bottom) A plot of the best fit of the hot glucose minimal model to a set of plasma tracer glucose data.

insulin is q3.FF = $I(t) - I_b$; it can be implemented as described in the previous example by associating the forcing function with the plasma insulin data. The constraints on the model parameters are $k(2,3) = p_2 \cdot S_I$ and $k(0,2) = p_2$. Finally, the glucose compartment, compartment 1, has a time-varying loss described by $k(0,1) = S_G + q2(t)$.

Figure 9 shows how the model would be implemented in SAAM II.

Input into compartment 1 is given by the input arrow ex1. As noted, the input consists both of the initial conditions, G_0, and the term $S_G G_b$, which is essentially a constant input throughout the experiment. There are basically two entries in the input dialogue box, both of which are equations. The first is the equation ex1 = G0; with a start and stop time equal to zero, this is equivalent to a bolus injection in this case equal to G0. The second equation is ex1 = SG*gss. In this equation, SG and gss correspond, respectively, to S_G and G_b in the differential equation dG/dt, where the latter is the glucose basal or steady-state concentration. The value for gss needs to be assigned. This equation is active for 240 minutes, the duration of the experiment, and results in a constant infusion.

The constraints on the model parameters and the description of the time-varying loss from compartment 1 are entered in the dialogue boxes associated with the respective transfer arrows. The time-dependent loss from compartment 1 is simply entered in the equation window as k(0,1) = SG + q2; this is representative of how to enter any nonlinear parameter in SAAM II.

The data used were glucose, insulin, and tracer concentrations. In the fitting process, weights are assigned to each datum; the weight is equal to the inverse of the variance. For glucose, a constant coefficient of variation of 1.5% was used. The insulin data, which act as a forcing function, do not have an assigned weight. Background insulin is also subtracted from the measured values. The error on the tracer data (in this case, the tracer was a stable isotope of glucose, [6,6-$^2H_2$]glucose) was deter-

mined by error propagation from the mass spectrometric direct measurements.[11]

The sample s1 in the classic minimal model is then associated with the glucose data. The sample s1 for the hot minimal model is associated with the tracer data. Insulin data are used to create the forcing function by joining sequential data by a straight line.

When the model has been fully specified as described and initial parameter estimates are entered, one can fit the model to the data. Figure 10 shows a representative fit of the classic model to a set of glucose data.

Implementation of the hot minimal model is very similar. In fact, the model on the drawing canvas looks exactly the same as the classic minimal model shown in Fig 9. The difference is that the model now represents Eq 6, which translates into different inputs and constraints on the parameters; However, it is important to note that the forcing function remains the same for both models.

In the hot minimal model, the input is a single bolus, since there is no term in the equation representing de novo glucose input. Thus, ex1 = G_0^*. The sample equation is now s1 = G*(t), ie, the sample is linked to the tracer data and not the cold glucose data. Finally, the constraints on the parameters are written in terms of the identifiable parameters S_G^*, p_2^*, S_I^*, and G_0^*: $k(0,1) = S_G^* + q2(t)$, $k(0,2) = p_2^*$, and $k(0,3) = p_2^* \cdot S_I^*$.

A typical fit to a set of tracer data by the hot minimal model is shown in Fig 10B. The difference between this and the fit shown in Fig 10A is the absence of input for the de novo production term, ie, the tracer data are decaying.

CONCLUSIONS

SAAM II is a state-of-the-art compartmental and numerical software modeling tool, developed by scientists with expertise in modeling theory and practice, computer science, and numerical and statistical techniques. The software is constantly being upgraded as new numerical and statistical theories are developed, and as needs from the user community demand.

After briefly describing the main features of the software and the algorithms used to estimate kinetic model parameters, we have given some examples to illustrate how many of the features in version 1.1 are used, and how to interpret the program output. We believe that the development of SAAM II has produced a software tool that is both easy to use and helpful for those designing and performing tracer and pharmacokinetic studies.

ACKNOWLEDGMENT

Development of SAAM II was made possible by Grant No. NCRR RR02176 from the National Institutes of Health. SAAM II is distributed by the SAAM Institute under a license agreement with the University of Washington. For more information, contact the Resource Facility for Kinetic Analysis homepage: http://weber.u.washington.edu/~rfka/ (or write to info@saam.com).

REFERENCES

1. Berman M, Weiss MF: The SAAM Manual. USPHS (NIH) Publication No. 78-180. Washington, DC, US Government Printing Office, 1978

2. Matthews CME: The theory of tracer experiments with ^{131}I labeled plasma proteins. Physiol Med Biol 2:36-53, 1957

3. Jacquez JA: Compartmental Analysis in Biology and Medicine. Ann Arbor, MI, 1996

4. DiStefano JJ: Noncompartmental vs. compartmental analysis: Some bases for choice. Am J Physiol 243:R1-R6, 1982

5. Bell BM, Burke JV, Schumitzky A: A relative weighting method for estimating parameters and variances in multiple data sets. Comput Stat Data Anal 22:119-135, 1996

6. Foster DM, Chait A, Albers JJ, et al: Evidence for kinetic heterogeneity among human low density lipoproteins. Metabolism 35:685-696, 1986

7. Cobelli C, Saccomani MP, Tessari P, et al: Compartmental model of leucine kinetics in humans. Am J Physiol 261:E539-E550, 1991

8. Parhofer KG, Barrett PHR, Bier DM, et al: Determination of kinetic parameters of apolipoprotein B metabolism using amino acids labeled with stable isotopes. J Lipid Res 32:1311-1323, 1991

9. Bergman RN, Iden ZY, Bowden CR, et al: Quantitative estimation of insulin sensitivity. Am J Physiol 236:E667-E677, 1979

10. Cobelli C, Pacini G, Toffolo G, et al: Estimation of insulin sensitivity and glucose clearance from minimal model: New insights from labeled IVGTT. Am J Physiol 250:E591-E598, 1986

11. Avogaro A, Bristow JD, Bier DM, et al: Stable-label intravenous glucose tolerance test minimal model. Diabetes 38:1048-1055, 1989