WinSAAM: A Windows-Based Compartmental Modeling System

Darko Stefanovski, Peter J. Moate, and Raymond C. Boston

Over the last 50 years, complex, dynamic, compartmental models have been used to describe and to make predictions on a host of pharmacokinetic, metabolic, and biological systems. Sophisticated modeling software is required to fit data to such models and to make predictions using these compartmental models. WinSAAM is one such modeling program. The purpose the current report is to describe the features of WinSAAM that make this program suited for modeling all manner of biological systems. We highlight new features, especially those that are unique to WinSAAM, and illustrate with examples how WinSAAM is used to construct models of metabolic systems, to simulate the effects of experiments on systems, and to fit models to data.

OVER THE LAST 50 years, complex, dynamic, compartmental models have been used to describe and make predictions on a large number of pharmacokinetic, metabolic, and biological systems. In particular, 2-pool compartmental models have been used to describe the linear kinetics of the disposition and metabolism of many pharmaceutical drugs.[1] Many biological systems can be described by compartmental analysis with nonlinear rate constants describing specific intercompartmental flows of material. For example, Bergman et al[2] developed a sophisticated nonlinear "minimal model" to describe how glucose and insulin interact with each other.

Making a prediction using these compartmental models necessitates solving a set of linear and/or nonlinear differential equations. This involves some fairly specialized algebra. For compartmental systems involving more than 3 compartments, the only practical approach involves the use of computers. Thus, there is a clear need for software to fit models to experimental data. In general, essential features of good modeling software should include capability to (1) simulate systems, (2) fit models to data, (3) provide measures of uncertainty of parameter estimates, and (4) to be user-friendly. WinSAAM is a modeling program with all of these features. It is a direct descendent of SAAM and CONSAM. The original SAAM (Simulation, Analysis, and Modeling) software, was first developed over 40 years ago, at the Laboratory of Experimental and Computational Biology, National Institutes of Health, Washington by Berman and Weiss.[3] Over the years, Boston and coworkers at the National Institutes of Health gradually developed SAAM into CONSAM,[4] SAAMII,[5] and lastly WinSAAM.[6]

There are many commercial programs available that are suited to compartmental modeling. However, it is not the intention of this report to review all of these modeling programs, or to compare their relative strengths and weaknesses. Indeed several publications have already compared available dynamic modeling software.[6,7]

A description of the theoretical aspects of WinSAAM, its novel modeling features, or examples of its use modeling biological systems, has not been previously published. This contribution is a survey of the general structure and features of WinSAAM that make it well suited for modeling biological systems. The information presented highlights new features unique to WinSAAM, and illustrates by example, WinSAAM modeling of metabolic systems, simulation of experiments on systems, and fitting models to data.

WinSAAM STRUCTURE AND LANGUAGE

There are 9 main parts or zones to WinSAAM. These include the terminal window (see Figs 2, 5, and 9), the text editor (see Fig 7), the charting system or plot window (see Fig 10), the spreadsheet manager, the project manager, the logger, the batch window, and the help system. The terminal window is the main interface between the user and WinSAAM. It is in the terminal window that the user types commands, uses pull down commands, or points and clicks on various icons so as to gain access to the different WinSAAM parts or zones. For example, the user may wish to move to the text editor, the spreadsheet, the charting window, or the help system.

The text editor is the main route by which the user modifies an input file that is used by the processing portion of the program. The text editor accepts and automatically manages data and parameters by means of a tab-delimited clipboard input and or tab-delimited file input.

The batch output window collects all output from the SAAM command into a single window. The WinSAAM log file enables collection of all user interactions with WinSAAM during a modeling session. Log files can be viewed, saved, appended, and overwritten.

The output spreadsheet tabulates the input data, model predictions, best-fit estimates of model parameters and their fractional standard deviations, the covariance matrix, and many data- and model-related calculations. The spreadsheet enables copying of results to the clipboard and the exchange of results with other statistical software, and the contents of the spreadsheet can be readily saved in EXCEL format.

The charting or plot window in WinSAAM provides the user with complete flexibility with regard to graphing data and model predictions. This system allows zooming of graphs, rescaling, logarithmic and linear displaying of either axis, line style, and plot point style assignment. Plots can be saved as files in EMF, WMF, JPG, or BMP formats or copied to the

From the School of Veterinary Medicine, University of Pennsylvania, Kennett Square, PA.

Submitted November 20, 2002; accepted February 24, 2003.

Address reprint requests to Raymond C. Boston, MD, New Bolton Center, University of Pennsylvania, 382 W Street Rd, Kennett Square, PA 19348.

0026-0495/03/5209-0041$30.00/0
doi:10.1016/S0026-0495(03)00144-6

clipboard and pasted into other programs (see Figs 1, 3, 6, and 8).

The WinSAAM Help system comprehensively describes the commands and conventions for using WinSAAM. It provides explanations on how to operate the software efficiently and it has sections to describe the structure and use of WinSAAM, modeling blocks, operational units, commands, demonstration models, a tutorial section, and a listing of frequently asked questions and their answers.

An important feature of WinSAAM that contrasts it with many other modeling systems is the modeless operation system in WinSAAM. This means that in terms of the activities supported by each WinSAAM window, the user has the flexibility to drive the sequence of a particular modeling investigation.

WinSAAM has more than 50 specific commands that invoke many different operations or procedures. The entire command system is very flexible in that a novice user may know and use perhaps only half a dozen commands, yet still be able to model complex systems. In contrast, a more experienced user may use a wider range of commands and be able to perform, for example, sophisticated postmodeling analyses.

WinSAAM retains the SAAM and CONSAM conventions regarding the nomenclature of operational units and parameters and the strict requirement that parameter names, initial estimates, and their boundary limits and data must all be entered into specific fields within the editor window. This requirement that parameter names and data all be entered in their specific fields constitutes what could be called the syntax of the "WinSAAM language." This WinSAAM syntax and the WinSAAM nomenclature describing modeling constructs together constitute the WinSAAM language. Using this WinSAAM language, a WinSAAM user can describe a complex model in a succinct and unambiguous fashion. Indeed, in a comparison of different software programs, the input scheme (files) required for WinSAAM were demonstrated to be simpler than those required for ACSL, MLAB, and Scientist.[6] Whereas the requirement for specific field entry was a source of user frustration in previous versions of SAAM and CONSAAM, the new WinSAAM text editor with appropriate tab delimited entry, greatly facilitates the writing of WinSAAM model schemes. In addition, a "help bar" located at the bottom of the editor window identifies the line number (of the input file) and the column number in which the cursor sits and the WinSAAM construct that can be entered in that specific field (see Fig 7).

WinSAAM COMPUTATIONS

In the routine application of WinSAAM to fit kinetic models to data, the WinSAAM user negotiates three processing steps: (1) model compilation, (2) model solution, and (3) model fitting. In the model compilation stage, the lexical structures are converted to appropriate equation and object form in preparation for their solution. In conjunction with this, extensive code analysis by WinSAAM will confirm that the specification of model and data are in good shape to advance the analysis. In the model solution step, the dependency equations are solved, then the state equations are solved, the linear system parameters are estimated, and, finally, the steady-state solutions (if needed) are found. In this step the automatic selection and tuning of the numerical integrator greatly reduces the need for the user to be skilled with numerical methods. WinSAAM supports seven highly refined numerical integrators, and will automatically select among three of these, the integrator that will result in the fastest and most accurate solution. Specifically, we automatically select amongst the following integrators: (a) Newton-Raphson (for explicit linear systems), (b) Chu Berman[8] (for implicit linear systems), and (c) a nonlinear variant of the Chu Berman integrator (for nonlinear systems). The other integrators can be manually selected for specific extreme situations or for corroborative purposes.

The linear or scale parameters K(j) (which will be discussed later), are given by:

$$QO(j) = K(j) \cdot QC(j, \theta) \tag{1}$$

where QO(j) represents the observed data in compartment j and QC(j,θ) are the calculated parameters [eg, θ represents the WinSAAM parameters P(j), L(i,j), DT(j), UF(j) upon which a solution is conditioned]. In WinSAAM, the K(j) are estimated using generalized or weighted least squares invoked in conjunction with the SOLVE command:

$$\hat{K}\,(J) = [QC' \cdot W \cdot QC]^{-1}\,[QC' \cdot W \cdot QO] \tag{2}$$

where (J) is the estimate of K(j). Equation 2 is written in matrix notation and the first part of the equation refers to an inverse matrix. The first term in this matrix refers is the transpose of QC, where QC denotes the solution to the state equations, the prime ($'$) denotes transpose, and W is the diagonal weight matrix. In the second matrix, QO denotes the observed values.

The nonlinear parameters are estimated using a variant of the Generalized Least Squares version of the Gauss Newton optimizer[9] that is invoked by using the ITER command. The processing steps associated with this are as follows. First, crude adjustments, $\delta\theta$, to the nonlinear parameters, θ, are found using:

$$\delta\theta = \left[\frac{\partial QC'}{\partial\theta} \cdot W \cdot \frac{\partial QC}{\partial\theta}\right]^{-1}\left[\frac{\partial QC'}{\partial\theta} \cdot W \cdot \delta Q\right] \tag{3}$$

where K is the transpose of the partial derivative vector of QC with respect to θ, and $\delta QC'/\delta\theta$ is the difference between observed and predicted values. Next, refinements, or improvements, to these raw adjustments are found by regressing the $\delta\theta$ on the data, viz:

$$C = [QO' \cdot W \cdot QO]^{-1}\,[QO' \cdot W \cdot \delta\theta] \tag{4}$$

where C is a factor applied to the crude adjustments to increase their utility. During this parameter estimation/iteration process, WinSAAM uses the specified upper and lower limits for parameters to appropriately confine the estimation of the parameters. To avoid instabilities in the data-fitting step, where necessary and appropriate, generalized matrix inversion is used, and this means that even when colinearity in the covariance matrix exists, the matrix inversion operation will not cause the program to fail. The fact that the estimation of linear system parameters is uncoupled from the nonlinear estimation step eases the process of nonlinear estimation substantially. Any other parameters including those that may have a linear context, eg, P(j), are also estimated iteractively using equations 3 and 4 above.

An alternative to the ITER command is the FIT command.

The FIT command uses a Nelder-Mead gradient-free optimization scheme to arrive at a best fit of model parameters.[10] The Nelder-Mead optimizer is not as susceptible as the 2-stage gradient optimizer to the starting estimates of parameters and it is more global in its search for a minimum. Thus, the FIT command may be useful to begin the fitting process or alternatively, once a "best fit" has been obtained by using the ITER command, the FIT command can be used to confirm that the fit has indeed converged on a global minimum. Thus, the variety of optimization procedures imbedded within WinSAAM make it fully concordant with the recommendations of Seber and Wild[9] regarding estimation tools appropriate for nonlinear modeling.

WinSAAM USES

WinSAAM is ideally suited to describing the kinetics of compartmental systems. A compartment is considered to contain material that is homogeneous or kinetically indistinguishable from other material in the compartment. In many cases the flow of material out of a compartment is proportional to the amount of material in the compartment, and this is known as first order kinetics. In WinSAAM notation, this is written as:

$$\frac{dF(1,t)}{dt} = -L(0,1) \cdot F(1) \qquad (5)$$

where F(1,t) [which is often written as F(1)], is the material in compartment 1 at time t, and L(0,1*)* is the fractional rate constant describing the flow of material from compartment 1 to the outside. In WinSAAM, L(i,j) is by default a first-order rate constant that describes the fraction of material in compartment j that moves into compartment i in unit time. Therefore, the existence of a particular L(i,j) in a WinSAAM input file serves the dual purpose of defining the existence of compartments i and j and also asserting that there is movement of material into compartment i from compartment j.

In addtion to fractional rate constants, WinSAAM also has an extensive array of other modeling constructs that facilitate the modeling process. These modeling constructs include initial conditions of compartments IC(j), compartment response F(j, t), user specified parameters P(j), scaling factors K(j), delay parameters DT(j), mass of compartments M(j) and steady-state input rates U(j), operational units (QO(j), QL(j), and QF(j)), forcing functions FF(j), time interrupt data elements TC(n), statistical weighting of data, and a facility to allow for function dependency. For mnemonic purposes, the nomenclature of WinSAAM constructs is somewhat acronymic. For example, "T" for time, "D" for delay, "M" for mass, "Q" for quantity, and "O" for observed. A number of these constructs will be discussed later and we provide some examples of their use. Furthermore, a complete description of all of the WinSAAM constructs can be found in the WinSAAM help section and in the book by Wastney et al.[6]

In general, the indices "i" or "j" associate a model construct or parameter with compartments "i" or "j." For example, F(1) describes the amount of material in question in compartment 1 at time "t". However, there is one exception to this convention. The general purpose parameter P(j), which is used ubiquitously in WinSAAM modeling schemes, is not associated with any specific compartment, and its model significance is assigned by the user. P(j) can be fixed in value, dependent upon another parameter, or adjustable, and it can be used in either a linear or nonlinear context. The delay construct DT(j) is a particularly powerful feature since it is adjustable and estimable and hence enables determination of lags or transit times (eg, a delay associated with gastric emptying, absorption, or intestinal transit) that may occur in some systems. Indeed, in one comparison of a number of modeling programs, only WinSAAM was found to adequately fit a model to data that involved a delay.[6]

In living entities, sometimes the kinetics of the metabolism of naturally occurring chemicals and of administered drugs may be described not by first-order kinetics, but may be described by a wide range of processes that have mathematical forms, which include: zeroeth order, second order, multi-reactant, Michaelis-Menten, inhibited elimination, bi-substrate "ping-pong" inhibition, simple inhibition, and allosteric binding. The functional dependency feature of WinSAAM is very useful since it can be used to modify or convert a L(i,j) into a nonlinear rate constant, which can model any of the nonlinear processes listed above.

SIMULATION OF BIOLOGICAL SYSTEMS AND EXPERIMENTAL PROTOCOLS

Some experimental protocols may involve not a single intervention, but a complex series of injections, infusions, filtrations, and other procedures. The host of model constructs described above also facilitates the representation of the most complex experimental protocols.

Simulation of Alcohol Metabolism

In the following example we describe how it is possible to construct a WinSAAM input scheme to simulate (model) the nonlinear kinetics of the metabolism of alcohol. In this example we also demonstrate how WinSAAM modeling constructs can be used to model a complex experimental protocol in which alcohol is consumed intermittently in 3 separate drinks. We assume that a male drinks three 100-mL glasses of wine of 10% alcohol concentration each over 15 minutes, and each drink is separated by a 20-minute period during which no wine is consumed. Figure 1 depicts the schedule of alcohol consumption.

Wagner et al[11] have shown that the rate of metabolism of alcohol can be described by:

$$\frac{dC}{dt} = \frac{-V_m \cdot C}{(K_m + C)} \qquad (6)$$

where C is the blood alcohol concentration (g/L) at time t, V_m is the maximum catabolism rate (0.18 g/L/h), and K_m is the Michaelis-Menten parameter (0.05 g/L). Figure 2 shows the WinSAAM input scheme (terminal window) used to simulate the experimental protocol described above. In contrast to the terminal window, in the edit widow, the input scheme can be displayed and edited. Starting in the first position of the first line of the WinSAAM input file is the heading A SAAM31. This heading necessarily initiates all WinSAAM input files. A problem title or description can be optionally entered in columns 31 to 72. The fourth line contains the heading "H PAR". This heading tells the WinSAAM program that WinSAAM constructs or parameters will occur on subsequent lines of the

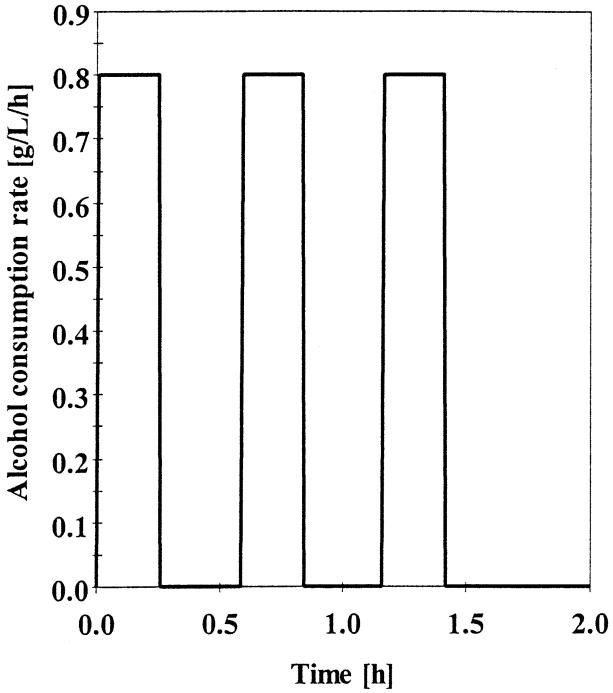


Fig 1. Experimental protocol for consumption of alcohol.

input scheme. The lines that are initiated with a "C" in column 1 are called comment lines, and are ignored by WinSAAM. However, comment lines are useful since they serve to annotate the input file so that the user can describe what is being modeled and the meaning of various parameters. In this example, we define the parameters P(1) to P(5) in lines 8 to 14 and in lines 16 to 20 we initialize these parameters as constants. The model assumes 100% alcohol absorption [P(5)=1, line 20] in a distribution space of 50 L [P(4)=50 line 19].Using WinSAAM constructs, equation 6 above can be rewritten as:

$$L(0,1) * F(1) = P(1) * F(1)/[P(2) + F(1)]$$

which simplifies to:

$$L(0,1) = P(1)/[P(2) + F(1)].$$

In this example, F(1) represents the amount of alcohol in compartment 1 at time "t". In WinSAAM, it is very easy to designate that a particular L(i,j) is not a linear or first-order rate constant, but is functionally dependent upon other pools or constituents. One way for the user to do this is shown on line 22 of the input scheme:

```
22: L(0,1)                    1    *G 1
```

L(0,1) was entered in field 1, followed by "1" in field 2 and then, the operator "*" followed by "G 1" in field 5. This line of code indicates that parameter L(0,1) is described by 1 * function G(1). In this example, line 24 of the input scheme contains the WinSAAM H DAT heading. This necessary heading informs the WinSAAM program that the following lines of text will relate to data. Line 26 of the input scheme contains the functional dependency equation G(1):

```
26: X G(1)=P(1)/(P(2) + F(1)).
```

Thus by using the WinSAAM "*" operator and WinSAAM constructs we are able to convert L(0,1) into a Michaelis-Menten rate constant. Note that the "X" in the first column of line 26 [ie, immediately before the G(1) dependency equation] is not part of the mathematics of the equation. The "X" simply informs WinSAAM that a time sensitive dependency equation follows.

In WinSAAM, there are 14 possible operators which include +, −, *, /, and 6 conditional operators (7, 8, 9, A, B, C). The use of the "8" operator will be described later and a full description of the use of all the operators is described by Wastney et al.[6] By using these operators and appropriate operands or dependency equations, there is no limit to the variety or types of reactions or processes that can be modeled.

In WinSAAM, there is generally more than one way to write an input file to model a particular system or aspect of the system. In this alcohol example we are faced with the decision of how to model a reasonably complex experimental protocol involving multiple drinks of alcohol being consumed at specified times. We could use a WinSAAM facility called a TC interrupt function. This facility allows separation of data associated with different phases of an experiment and the resetting of parameter values to accommodate aspects of the model that may change in relation to different phases of the experiment.

However, we can also succinctly describe the alcohol intake pattern described above using the WinSAAM QO function. In this specific case, QO(2) serves as a simple on/off switching function for a continuous infusion. However, QO functions are very versatile since they can also be used to describe for example any combination of continuous, ramped infusions or even simple injections, all occurring at specified times.

A primary aim of modeling is to obtain model predictions, and this is very easy to do with WinSAAM. In this alcohol example (see Fig 2) we request predictions of the alcohol consumption rate (experimental protocol), the cumulative alcohol delivery and the blood alcohol concentration till 4 hours, and using the WinSAAM charting system obtain the graphs shown in Figs 1 and 3. Further discussion on the use of WinSAAM for modeling alcohol metabolism can be found in Wastney et al.[6]

FITTING MODELS TO DATA

A 2-compartment model with transfer between a central and peripheral compartment and elimination from the central compartment is often used to describe the kinetics of the distribution of administered pharmaceutical drugs.[1] Usually a drug may be given either as a bolus dose, or by a constant infusion, or even by a "priming" dose followed by a constant infusion. Figure 4 presents the conventional way for depicting an open 2-compartment model. Such diagrammatic representations of systems greatly assist the user to implement models within WinSAAM. The 2-pool model is used so often that its underlying mathematics may sometimes be taken for granted and there is ample evidence in the pharmacokinetic literature that many researchers have applied inappropriate assumptions to the 2-pool model and obtained erroneous estimates of model parameters. Let us consider the experimental protocol where a priming dose (1,000 μg/kg) of drug X was administered to a

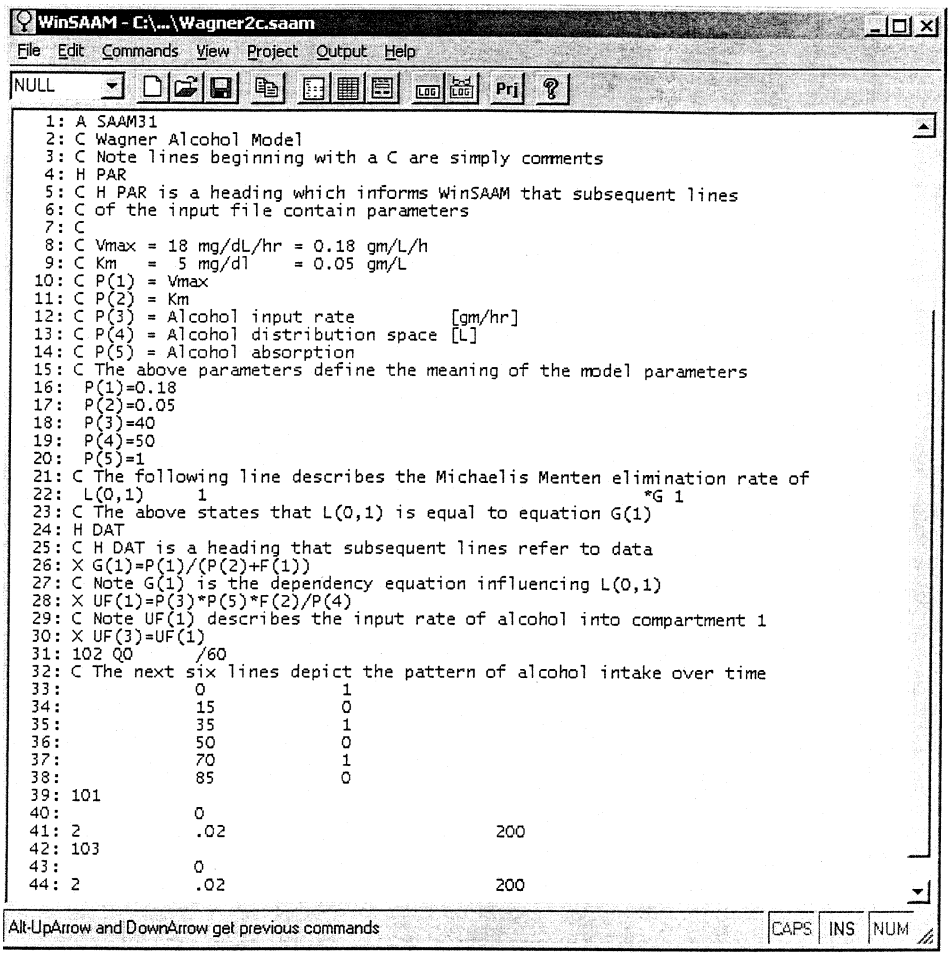


Fig 2. WinSAAM input scheme (shown in the terminal window) for Wagner's alcohol model.

subject and then the drug was infused into the subject for 15 minutes at a constant rate (300 μg/min). In this particular experiment, the researcher took blood samples at 5, 10, 15, 18, 21, 24, 27, 30, and 35 minutes following the initial injection of drug. The protocol for this experiment is easily entered into a WinSAAM input scheme (see Fig 5). In this case, the injection is simulated by means of the initial condition construct IC, and, as in the alcohol example shown above, the infusion is simulated by means of the QO construct.

WinSAAM input schemes describing tracer kinetics or drug metabolism generally include a scaling constant K(1). This scaling factor takes into account the fact that data are usually expressed in terms of concentrations, but in WinSAAM, it is usually the total amount of substance (tracee) or drug within a compartment that is modeled. K(1) is inversely related to the volume of distribution of the primary or central compartment (Vd_1) into which the tracer or drug was administered. In experiments involving simply an injection of tracer or drug, this K(1) is the time zero y intercept. In this particular protocol, blood samples were not collected until 5 minutes after injection of the drug and without making any assumptions, it is difficult to estimate K(1) directly from the data. However, in this protocol, drug was infused at a constant rate and appears to have reached a steady-state concentration by 15 minutes. At steady-state, the flow of drug into the patient is equal to the

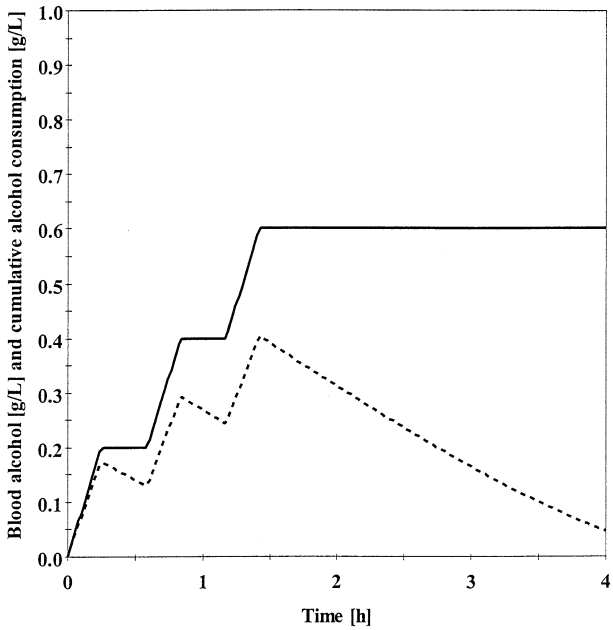


Fig 3. Cumulative delivery of alcohol (g/L) (—) and the model predicted blood alcohol concentration (g/L) (---).

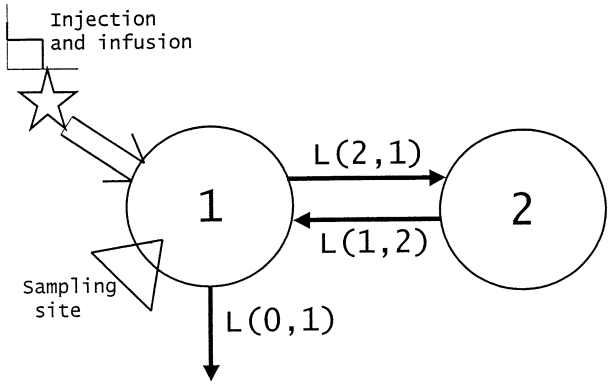


Fig 4. Diagrammatic representation of a 2-compartment model. L(2,1) represents the fractional rate constant describing movement of drug from the site of injection to another body compartment, while L(1,2) is the fractional rate constant representing the reverse movement. L(0,1) represents the fractional rate constant describing elimination of the drug. The star and double arrow represent the injection and infusion while the triangle represents a sampling site.

flow of drug out of the patient and this can be expressed as follows:

$$P(2) = P(4) * P(8) * L(0,1) \quad (7)$$

where P(2) is the steady-state infusion rate of drug into the patient (300 μg/min), P(4) is the volume of distribution of the central compartment, P(8) is the steady-state concentration of drug in the central compartment and L(0,1) is the fractional elimination rate of drug from the central compartment. By rearranging equation 7 and using the fact that K(1) is inversely related to the volume of the central compartment, we have:

$$K(1) = P(8) * L(0,1)/P(2).$$

Thus, if we make the assumption that the steady-state plasma concentration of drug had indeed been reached by 15 minutes, we can include the above equation in the WinSAAM input scheme (see Fig 5, line 18) and this serves as a constraint on the magnitude of the L(0,1) parameter and it helps the fitting algorithm converge on a feasible solution. We can then use the drug concentration at 15 minutes as an initial estimate of the steady-state concentration of drug P(8). This is shown in Fig 5, on line 24 of the input scheme:

24: P(8) 4.127505E+03 0.0E+00 5.000E+03

On line 24 of the input scheme, the parameter name, P(8), is entered in the first field followed by the initial estimate or value of P(8) in the second field, the lower limit of the value of P(8) is entered in the third field, the upper limit is entered in the fourth field. Similarly, for L(0,1), L(2,1), and L(1,2) on lines 25, 26, and 27, respectively, we enter the initial estimate of each parameter, its lower limit and upper limit in the appropriate fields.

The data relating to the concentration of drug in plasma (compartment 1) are entered in the input scheme under the H DAT heading. In this case, the concentrations of drug in plasma range over several orders of magnitude. To ensure that the model fitting algorithm will produce a "best fit" that can in a relative sense, equally well predict both high and low plasma concentrations of drug, we weight the data in terms of the fractional standard deviation (FSD) of each datum. However, with WinSAAM, if it is appropriate to the data being modeled, we may alternatively choose a weighting scheme based on either the standard deviation (SD), the square root of the standard deviation (RQO) or a constant weight (WT).

Once the WinSAAM input scheme to describe this model has been constructed, the WinSAAM modeler need only invoke the following simple sequence of commands to obtain the best-fit estimates of model parameters. First, the modeler must type into the terminal window (or use the pull down command menu) the command: "DECK". This command checks for syntactical errors in the input file and compiles the input file in a form suitable for processing by the WinSAAM fitting algorithm. The next essential command is "SOLV". As described earlier, this command uses initial estimates of parameter values to calculate "predicted" values of the function being modeled. This command also produces an objective function that is in fact the sum of the squares of the differences between the actual data observations and the current model predictions. When, as in this case, it is the intention of the modeler to obtain best estimates of model parameter values so that the model will describe the variation in the plasma drug concentration over time, we use the ITER command. As described earlier, this command invokes a variant of the Gauss-Newton algorithm that seeks to minimize the objective function. Fitting the model to data may require several invocations of the ITER command. Output following each command includes the sum of squares of the objective function(s), the percent reduction in the sum of squares, adjustment details of the parameters whose values are most altered during the iteration. A final best-fit combination of model parameters has been achieved when WinSAAM responds that it is not possible to further improve the fit. The best estimates of the model parameters and their fractional standard deviations are then obtained by typing the command FSD(i), and a graph of the data and model predictions can be obtained by invoking the "PLOT" command.

In this particular patient, the model parameters were all well identified (as judged by low fractional standard deviations for each parameter) and the model predictions closely coincide with the observed data (Fig 6). However, in some subjects, there may be a precipitous decline in plasma drug concentrations immediately after cessation of the drug infusion, and this may present some difficulties for accurately estimating model parameters, especially if there are few samples during this period. In some experimental protocols, there may be few samples taken during the constant infusion period and some pharmacokineticists ignore the data obtained during the initial infusion period and instead focus on analyzing the data obtained after cessation of the infusion of the drug. Unfortunately, many make the mistake of assuming that at the instant that drug infusion ceases, it is only the first compartment that contains drug and they analyze the data as if it resulted from an injection into the primary compartment made at the time of cessation of drug infusion. If we assume that the drug was infused for a sufficient period so that steady state conditions had been reached by the time the drug infusion was stopped, then we can indeed focus our investigation on the disappearance of drug following cessation of the drug infusion, provided we impose

Fig 5. WinSAAM input scheme (shown in the terminal window) to describe drug disposition from a 2-compartment model.

constraints on the model that correctly reflect the amounts of drug present in both the central and peripheral compartments at the instant the drug infusion was stopped.

The WinSAAM input scheme (editor window) for this analysis is shown in Fig 7. Note that the input scheme is in many ways similar to that shown in Fig 5. However, in this case, since we are modeling the system from the instant the infusion is stopped, we must specify the initial conditions of both compartments. For compartment 1 (the central compartment), we can assume that the initial concentration is equal to the steady state concentration P(8), and that the concentration of the drug at 15 minutes is a good initial estimate of this parameter. However, the observed drug concentration at 15 minutes is a datum and like all the other data has a degree of uncertainty or error associated with it. In recognition of this we can again

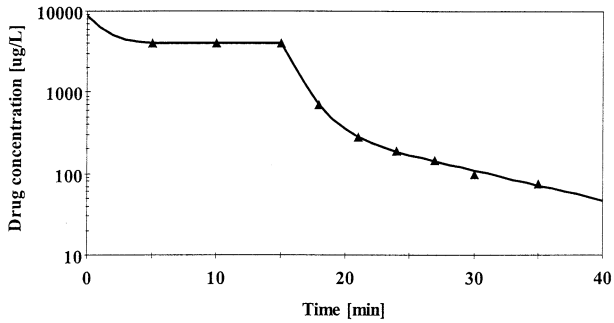


Fig 6. Modeling the disposition of drug following an injection and a 15-minute infusion. The triangles are observed data and the solid line is predicted by the 2-compartment model. The model parameters and their (fractional standard deviations) are as follows: P(8) 4.128E+03 (2.804E−02); L(0,1) 6.478E−01 (4.716E−02); L(2,1) 6.550E−02 (9.498E−02; L(1,2) 9.386E−02 (8.038E−02).

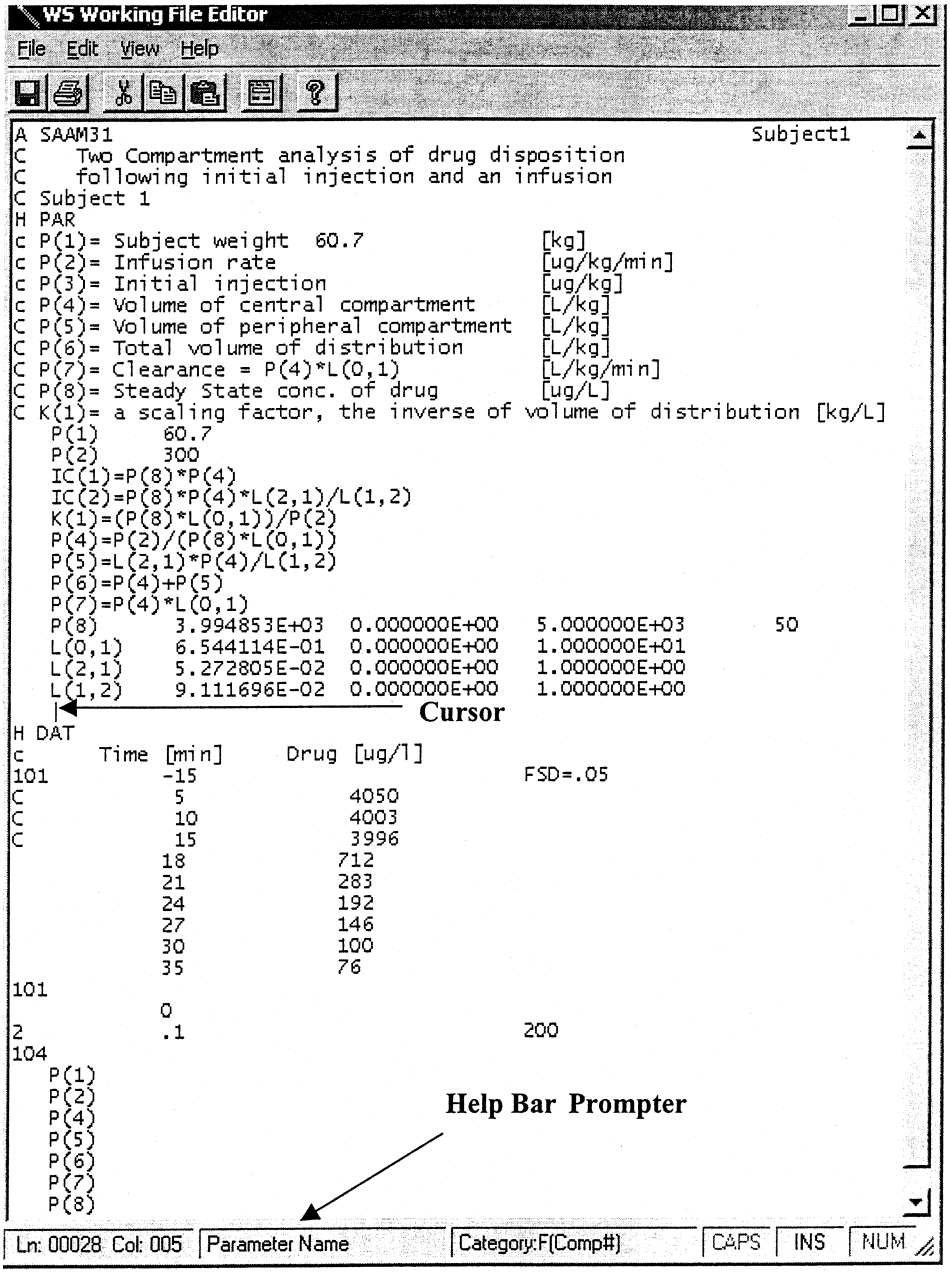


Fig 7. A WinSAAM input scheme (shown in the editor window) to model drug disappearance following cessation of infused drug. Note, the "help bar" identifies the line number (of the input file), the column number in which the cursor sits and the WinSAAM construct that can be entered in that specific field.

make P(8) an adjustable parameter (by inserting upper and lower limit boundaries), but this time we assign it a reasonable standard deviation. This is shown in Fig 7 where on the same line where we entered P(8), we have also entered in field 7 an estimate of the standard deviation (in this case 50). We can then specify the amounts of drug present in the central IC(1) and peripheral IC(2) compartments at the instant the infusion was stopped as:

$$IC(1) = P(8) * P(4)$$

$$IC(2) = P(8) * P(4) * L(2,1)/L(1,2) \quad (8)$$

Using this analysis we obtain estimates of model parameters (see Fig 8) that are not statistically different from those obtained in the earlier analysis. However, the first approach generally produced lower estimated fractional standard deviations of the parameters. Thus, in this example, the first approach is to be preferred because it estimates model parameters by making use of all of the available data. However, if for example data were only available from time 15 minutes onwards, and if it were known a priori that a constant infusion of this drug would result in a steady-state concentration of drug in less than 15 minutes, then we could use the second approach. Thus, the second example demonstrates that with WinSAAM, we can easily make use of a priori or theoretical knowledge about the system, impose such information as constraints upon the parameters being estimated, and thereby obtain good estimates of those parameters.

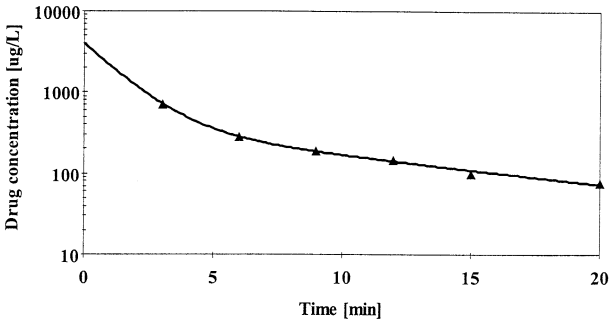


Fig 8. Modeling the disposition of drug following an injection and a 15-minute infusion. The triangles are observed data and the solid line is predicted from the 2-compartment model. The model parameters and their (fractional standard deviations) are as follows: P(8) 3.995E+03 (1.251E−02); L(0,1) 6.544E−01 (4.252E−02); L(2,1) 5.273E−02 (1.202E−01); L(1,2) 9.112E−02 (8.358E−02).

Modeling Glucose and Insulin Kinetics

Diabetes, a disease that can cause blindness, kidney failure, neuropathy and heart disease currently affects over 16 million people in the United States.[12] It is primarily caused by a malfunction of the mechanisms that control the metabolism of glucose. The metabolism of glucose is controlled by circulating plasma glucose per se and by insulin that is secreted from the pancreas in response to elevated levels of plasma glucose. In order to be able to describe mathematically how glucose and insulin together control the production and disposal of glucose in the body, Bergman et al[2] developed the now famous 1-compartment minimal model. With the prevalence of diabetes now reaching epidemic proportions, it is not surprising that interest in the minimal model continues to grow with more than 50 reports being published each year on issues related to the minimal model.[12] The parameters of the minimal model are determined using data from a standard in vivo glucose tolerance test (IVGTT). In the IVGTT a patient is intravenously administered a bolus of glucose (300 mg/kg) followed 20 minutes later by an intravenous infusion of insulin (20 mU/kg), and frequent blood samples are collected over the 3 hours following the glucose injection. The Bergman minimal model is described by 2 equations:

$$\frac{dG(t)}{dt} = -S_G[G(t) - G_b] - x(t) \cdot G(t) \quad G(0) = G0 \qquad (9)$$

$$\frac{dx(t)}{dt} = -p2\{x(t) - S_I[I(t) - I_b]\} \quad x(0) = 0 \qquad (10)$$

where G(t) and I(t) are glucose and insulin concentrations in plasma respectively and G_b and I_b are the basal concentrations of glucose and insulin in plasma respectively. G0 is the model-predicted value of the glucose concentration at time zero, while x(t) is insulin action and p2, is a parameters to describe the removal rate of insulin from interstitial fluid. S_I or insulin sensitivity (the capacity of insulin to promote glucose disposal) and S_G or glucose effectiveness (the capacity of glucose to mediate its own disposal) are parameters to describe the net removal rate. For a discussion on the use of WinSAAM for fitting the minimal model to data and a WinSAAM input file to accomplish this, the reader is referred to chapter 15 of the book *Investigating Biological Systems Using Modeling*.[6]

A variation of the IVGTT is the tracer-labeled intravenous glucose tolerance test (HIVGTT) in which the subject is injected simultaneously with both nonlabeled glucose (300 mg/kg) and a glucose tracer (g0; a "hot" radioactive or stable isotope). The purpose of the HIVGTT is to attempt to quantify the dual effects of insulin, ie, promotion of the disposal of glucose from the blood and a reduction in the rate of de novo synthesis of glucose. In the hot minimal model, developed by Cobelli et al,[13] the differential equation describing the disappearance of labeled "hot" glucose (g) is similar to equation 9 except that the term involving the nonlabeled basal glucose is omitted:

$$\frac{dg(t)}{dt} = -[S_G^g + x^g(t)] \cdot g(t) \quad g(0) = g0 \qquad (11)$$

$$\frac{dx^g}{dt} = -p^g\{x^g(t) - S_I^g[I(t) + I_b]\} \quad x^g(0) = 0 \qquad (12)$$

where the superscript "g" refers to parameters associated with labeled glucose. In contrast to the "cold" minimal model where S_G and S_I are indices of the net removal rate of glucose, in the "hot" minimal model, S_G^g and S_I^g are indices that measure respectively the effect of glucose and insulin on only the disposal rate of glucose while p^g has similar meaning to P2 in the cold minimal model. Despite initial promising findings with the "hot" minimal model, when deconvolution procedures were employed to estimate endogenous glucose production by using parameters from the hot minimal model and data from the cold IVGTT, unphysiological patterns were obtained. This probably occurred due to a failure of the monocompartmental model to adequately represent the kinetics of the early mixing of glucose in the extracellular fluid and other tissue spaces.[14] A 2-compartment model has recently been shown to overcome this problem and it can produce plausible estimates for endogenous glucose production.[14,15] The equations for the 2-pool model are as follows:

$$\frac{dg_1(t)}{dt} = -[k_p + \frac{F_{01}}{V_1 \cdot G_1(t)} + k_{21}] \cdot g_1(t) + k_{12} \cdot g_2(t)$$
$$g_1(0) = g0 \qquad (13)$$

$$\frac{dg_2(t)}{dt} = k_{21} \cdot g_1(t) - [k_{02} + x(t) + k_{12}] \cdot g_2(t)$$
$$g_2(0) = 0 \qquad (14)$$

$$\frac{dx(t)}{dt} = -p2 \cdot \{x(t) - S_k[I(t) - I_b]\} x(0) = 0$$
$$g(t) = \frac{g_1^{(t)}}{V_1} \qquad (15)$$

where g_1 and g_2 denote labeled glucose in the first (accessible pool which corresponds to extracellular fluid) and the second (slowly equilibrating or exchangeable) compartments, respec-

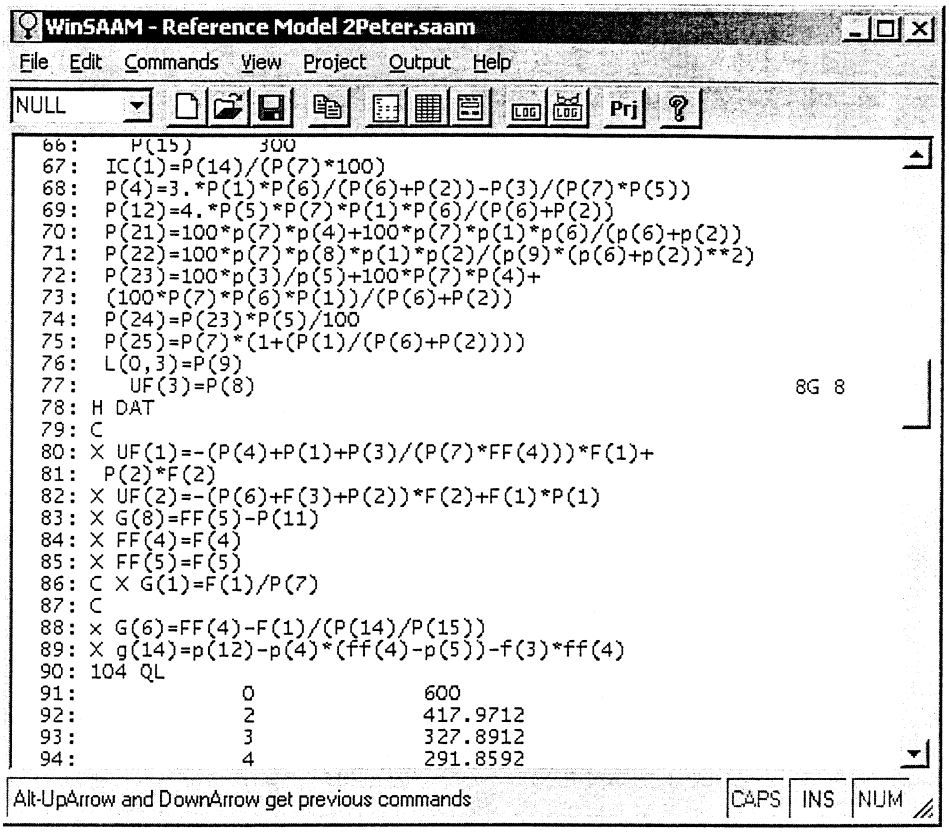


Fig 9. Part of the WinSAAM input scheme (shown in the terminal window) for modeling the 2-pool "hot" glucose model.

tively; G_1 is cold glucose in the accessible pool; x(t) is the insulin action; I(t) and Ib are plasma insulin and plasma basal insulin; V_1 is the volume of the accessible pool; F_{01} is the constant component of glucose disposal; k_p, k_{12}, k_{21}, and k_{02} are first-order rate constants describing glucose kinetics; and p2 and S_K are parameters describing insulin action.

The Bergman 1-compartment minimal model actually involves 2 compartments, (plasma "cold" glucose and plasma insulin), whereas the 2-pool hot glucose model actually involves 3 compartments (accessible "hot" glucose, slowly equilibrating "hot" glucose and "remote" insulin). Despite the small number of compartments, these models are all relatively complex in that they have many parameters and are nonlinear in nature. However, all of these models can be readily implemented in WinSAAM. Indeed, WinSAAM has a number of features, which greatly facilitate the modeling of these systems.

In many cases we may wish to impose constraints on parameter values or relationships between parameters. Indeed, in order to guarantee parameter identifiability and ensure their model was physiologically sound, Caumo and Cobelli[14] constrained the basal steady-state insulin-dependent glucose disposal to be 3 times insulin-dependent glucose disposal. Using WinSAAM, it is easy to include such contraints in a model and this particular constraint is shown in Fig 9, line 68.

In the alcohol and drug models described earlier, we used the WinSAAM L(i,j) construct to implicitly define the differential equations describing the movement of tracer or drug between compartments. However, in some circumstances a modeler may wish to define a system explicitly by means of differential equations, and this can be carried out quite easily using the WinSAAM UF(j) construct. UF(j) defines a component of the rate of input into compartment j. UF(j) allows the user to define highly complex system inputs. For example, equation 13 above can be directly translated into WinSAAM constructs as follows:

$$UF(1) = -[P(4) + P(1) + P(3)/P(7)*FF(4))] * F(1) + P(2) * F(2) \quad (16)$$

where UF(1) equates to $dg_1(t)/dt$, F(1), F(2), and FF(4) are functions representing $g_1(t)$, $g_2(t)$, and $G_1(t)$, respectively, and P(1), P(2), P(3), P(4), and P(7) correspond to the appropriate constants in equation 13.

In equations 13 and 15 above, Caumo and Cobelli[14] used both $G_1(t)$ (the cold plasma glucose) and I(t) (the plasma insulin) as functions in time, to drive the hot glucose model. However, in the HIVGTT, plasma samples are only taken at discrete points in time and for the hot 2-pool model to work, a way must be found to turn these discrete measurements into continuous functions. This is easily achieved in WinSAAM by using the QL operation. In WinSAAM, all data are entered into the WinSAAM input scheme under a H DAT heading (see Fig 9). Under the H DAT heading, a number in column 2 and 3 defines the component number to which the subsequent data belong. In this example, cold glucose data are associated with, or rather defined as belonging to compartment 4. When QL is entered into column 5 and 6 on the same row as the compartment number, this invokes the QL operation. The QL operation

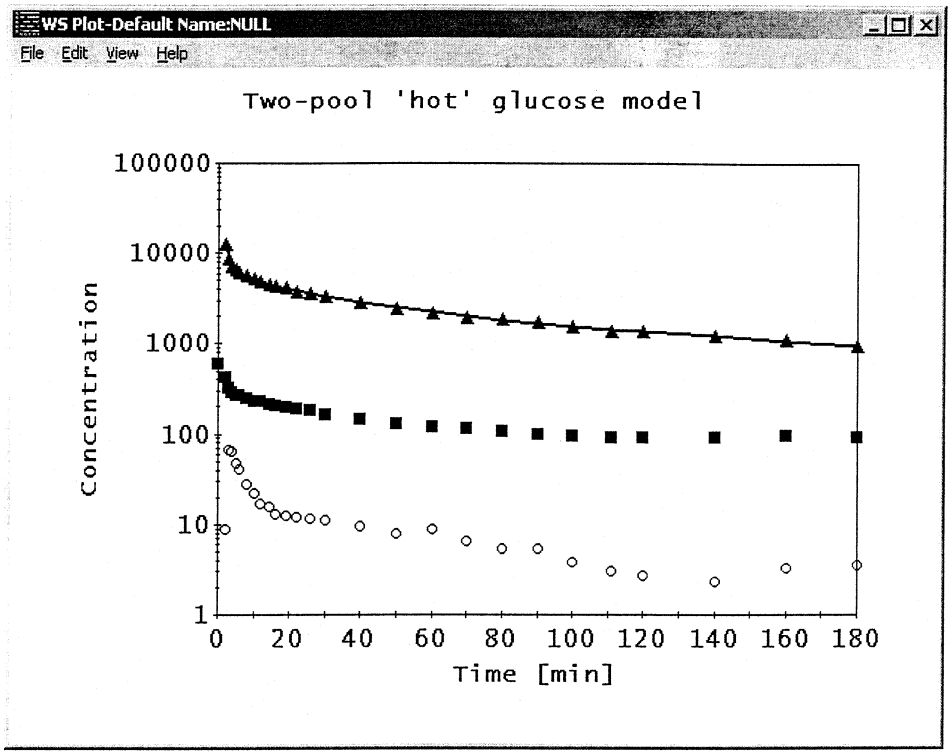


Fig 10. Prediction of concentrations of "hot" plasma glucose (shown in the charting window) using the 2-pool "hot" glucose model. (▲) "Hot" glucose (cpm/mL); (—) predicted line of best fit; (■) "cold" plasma glucose (mg/dL); (○) plasma insulin (mU/L).

produces a forcing function FF(j,t) that is based on a linear interpolation of the data associated with component J. Thus in equation 16 above (see also Fig 9), FF(4) represents a continuous function describing cold plasma glucose. Equation 15 above also implies that the movement of insulin from plasma to the remote site (where insulin has its action) occurs (only) if plasma insulin is greater than the basal concentration of insulin in plasma. Basal insulin is taken as the plasma insulin concentration immediately before the start of the IVGTT. Therefore, when modeling remote insulin, there is a requirement that the term [I(t) − Ib)] can only take non-negative values. This can be implemented in WinSAAM by using the functional dependency feature of WinSAAM and the "8" operator. This is shown on line 77 of the input scheme (see Fig 9):

77: UF(3)=P(8) 8G8

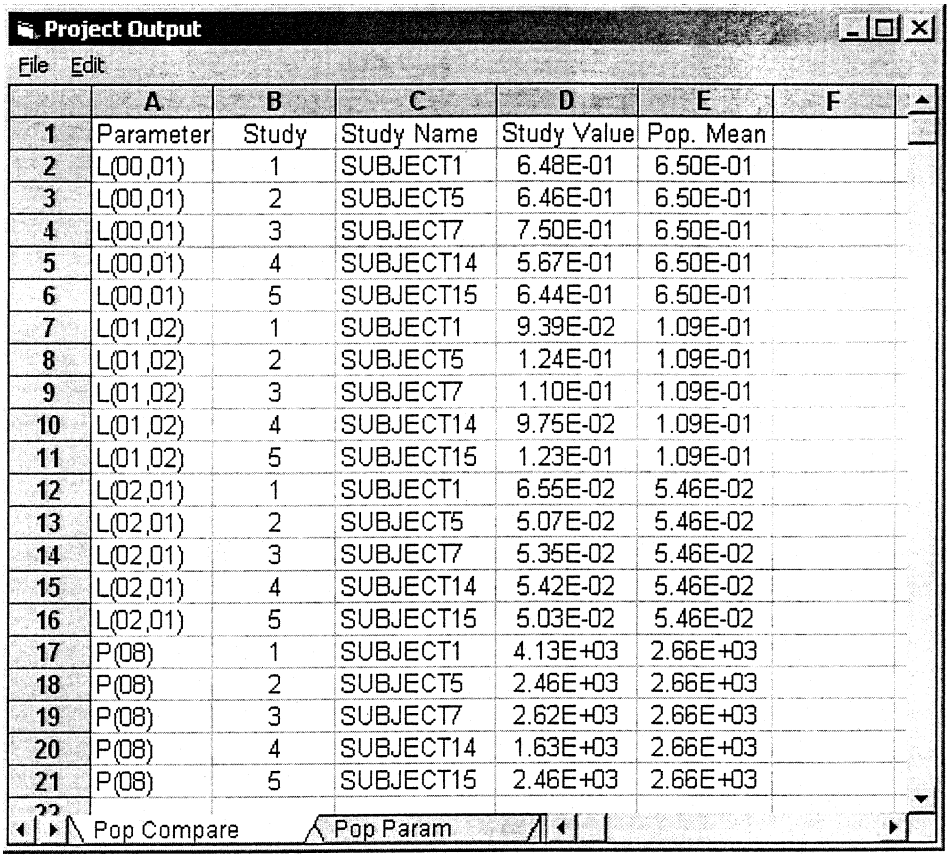
In this case UF(3) equates to dx(t)/dt, P(8) is a constant, G 8 equates to [I(t) − Ib], and the "8" immediately before the G 8 is the conditional operator. This operator acts as a switch which has the following effect: if G 8 is greater than zero then UF(3) = P(8) * G(8) else 0. This type of conditional switch is particularly useful for modeling systems where a particular reaction or rate constant may be switched on only when a drug or metabolite reaches a critical threshold value. An example of a WinSAAM prediction of "hot" plasma glucose, based on the "hot" 2-pool model, is shown in Fig 10.

Project Management

A new and perhaps one of the most advanced features of WinSAAM is its project management feature. This feature enables the user to obtain population estimates of kinetic parameters. This type of analysis is becoming important in pharmakockinetics since it is increasingly recognized that the metabolism of drugs may be very different in various target populations. Typically, an experimenter may have 2 groups of individuals; a control group and a second group of individuals upon which he has imposed some experimental treatment. The experimenter may then do a tracer or metabolic study on individuals in each group with the intention of determining if treatment influences the kinetic parameters. The project facility in WinSAAM is especially suited to the analysis of data from a group of subjects in identically replicated experiments and in which a compartmental model of fixed topology is used to analyze data from individual subjects within the group. WinSAAM's forte is in data-rich (meaning each subject's model is identified), as opposed to data-sparse situations.

The "project" facility allows 2 options. In the first option, the user may simply by pointing and clicking, select a series of the individual WinSAAM input files of all of the subjects in a particular group and then run these files in WinSAAM. This results in a spreadsheet output file that contains all of the individually fitted parameter values for each subject as well as the fractional standard deviations for each parameter. This facility allows results to be quickly scanned to help identify errors in data and individuals who are "outliers" and who may not necessarily fit within a particular population.

In the past, when making population estimates of model parameters, most researchers either simply averaged parameter estimates from individual studies or fitted a model to aggregate data. These approaches fail to recognize that some individuals may not necessarily fit within a given population. Furthermore, in many situations these approaches may lead to spurious estimates since individual studies are likely to produce differingly precise estimates of model parameters and because within



Project Output

File Edit

	A	B	C	D	E	F
1	Parameter	Study	Study Name	Study Value	Pop. Mean	
2	L(00,01)	1	SUBJECT1	6.48E-01	6.50E-01	
3	L(00,01)	2	SUBJECT5	6.46E-01	6.50E-01	
4	L(00,01)	3	SUBJECT7	7.50E-01	6.50E-01	
5	L(00,01)	4	SUBJECT14	5.67E-01	6.50E-01	
6	L(00,01)	5	SUBJECT15	6.44E-01	6.50E-01	
7	L(01,02)	1	SUBJECT1	9.39E-02	1.09E-01	
8	L(01,02)	2	SUBJECT5	1.24E-01	1.09E-01	
9	L(01,02)	3	SUBJECT7	1.10E-01	1.09E-01	
10	L(01,02)	4	SUBJECT14	9.75E-02	1.09E-01	
11	L(01,02)	5	SUBJECT15	1.23E-01	1.09E-01	
12	L(02,01)	1	SUBJECT1	6.55E-02	5.46E-02	
13	L(02,01)	2	SUBJECT5	5.07E-02	5.46E-02	
14	L(02,01)	3	SUBJECT7	5.35E-02	5.46E-02	
15	L(02,01)	4	SUBJECT14	5.42E-02	5.46E-02	
16	L(02,01)	5	SUBJECT15	5.03E-02	5.46E-02	
17	P(08)	1	SUBJECT1	4.13E+03	2.66E+03	
18	P(08)	2	SUBJECT5	2.46E+03	2.66E+03	
19	P(08)	3	SUBJECT7	2.62E+03	2.66E+03	
20	P(08)	4	SUBJECT14	1.63E+03	2.66E+03	
21	P(08)	5	SUBJECT15	2.46E+03	2.66E+03	

Pop Compare | Pop Param

Project Output

File Edit

	A	B	C	D	E
1	Parameter	Pop. Mean	Est. Error	Pop. STD	
2	L(00,01)	6.50E-01	2.61E-02	5.07E-02	
3	L(01,02)	1.09E-01	5.44E-03	9.30E-03	
4	L(02,01)	5.46E-02	1.87E-02	4.13E-02	
5	P(08)	2.66E+03	3.62E+02	8.05E+02	

Pop Compare | Pop Param

Fig 11. Spreadsheet output from population analysis of drug disposition. The population compare sheet allows comparison of model parameters values from each subject in the population. The project parameter sheet provides the maximum likelihood unbiased estimates of population parameter means and their population standard deviations.

an individual study, estimates of model parameters may not be independent of one another. The WinSAAM population modeler recognizes the deficiencies associated with employing either averaging the parameters or fitting a single model to the aggregate data by: (1) identifying individuals that do not fit within the population under study, and (2) deflating the contribution of the individuals to the estimation of the population parameters if they have imprecise or poorly defined parameters.[16-18]

The project management tool in WinSAAM is distinct among programs that estimate population kinetic parameters in that it employs a global 2-stage procedure based on the use of the maximum likelihood principle that produces unbiased estimates of population parameters.[16,18] Recently, the WinSAAM approach for population parameter estimation was compared with that incorporated into NONMEM in the context of norepinephrine kinetics and the parameter dispersions from WinSAAM were found to be less biased than those derived using NONMEM.[6]

To utilize the population option, the WinSAAM user, need simply select the WinSAAM input files of individuals from a particular group and, by toggling a selection button, opt for a population analysis. This results in a spreadsheet output file that contains nine different worksheets. There are worksheets that contain (1) the data from each study; (2) the model parameters and their fractional standard deviations as determined for each individual study; (3) the covariance matrices of the parameters in each study; (4) the correlation matrices for the parameters in each study; (5) the partial derivatives of the solutions with respect to the parameters in each study; (6) the covariance matrix of the population parameters; (7) the correlation matrix of the population parameters; (8) the population comparison

worksheet in which the individual study parameters are listed alongside their corresponding population mean parameters; and (9) the population parameter worksheet which contains the unbiased estimates of the population mean and population fractional standard deviations for each of the model parameters. Figure 11 shows the population comparison worksheet and the population parameter worksheet obtained when we conducted a population analysis of the drug kinetics of 5 individual subjects, including the subject whose data are shown individually analyzed in Fig 5.

The results from the WinSAAM project management environment can readily be exported to EXCEL or other Windows compatible spreadsheets for post processing or productivity presentations.

PRESENT AND FUTURE DIRECTIONS

WinSAAM is not standing still. The WinSAAM developmental team is constantly adding new features to WinSAAM, to ensure that WinSAAM remains at the cutting edge of modeling technology. The WinSAAM website WWW.WinSAAM.com has the latest information on WinSAAM, contact details and an installable version of WinSAAM can be downloaded from the website.

We are actively pursuing hybrid program development based on the SAAM kernel. The latest advance is the development of computer programs dedicated to modeling specific biological/metabolic systems. An example is MinMod Millenium, a program that automatically models data to Bergman's minimal model. This particular program is particularly suited to users with no modeling skills since it is completely automatic. It utilizes SAAM technology, an expert system to initiate and guide the modeling procedure and a user-friendly interface. We foreshadow that this program is but a forerunner of a suite of similar programs. Another offshoot program that has just been developed is AKA-Glucose. This program also utilizes SAAM technology, an expert system and a user-friendly interface. This program allows either automatic fitting of data to the Bergman minimal model, or the user can intercede and direct the modeling procedure. The main novel and potentially enormously useful feature of AKA-Glucose is that it incorporates a database. A researcher can use AKA-Glucose to store individual demographic data on thousands of subjects, to analyze the IVGTT on these same subjects, to stratify IVGTT studies on the basis of demographic and MinMod parameters and carryout other epidemiological analyses.

CONCLUSIONS

This report is a survey of WinSAAM and how it can be used for simulation and parameter estimation of linear and nonlinear compartmental models, with emphasis on new WinSAAM features for nonlinear parameter estimation and population kinetic data analysis. Topics include an overview, WinSAAM structure and language, WinSAAM computations, WinSAAM uses including examples on the modeling of alcohol metabolism, drug metabolism, and glucose/insulin kinetics. In conclusion, SAAM a product of 1950s was designed to solve the kinetic problems of the second half of the twentieth century. WinSAAM, the son of SAAM, has been designed to solve the kinetic challenges of the 21st century. The retention in WinSAAM of the proven mathematical algorithms of SAAM, the inclusion in WinSAAM of complete compatibility with Windows technology, the myriad of modeling constructs, the population modeling capability and the other features described above indicate why WinSAAM is the appropriate tool for these challenges.

ACKNOWLEDGMENT

Development of WinSAAM was made possible by a series of grants: CAN78333487, CAN38322423 and CA82707 from the National Institutes of Health. WinSAAM would not have been possible if it were not for the seminal work of the late Dr Mones Berman in developing SAAM, the vision and encouragement of the late Dr Loren Zech, and the innovative computing skills of Dr Peter Grief. We are grateful to Dr J. Maizel, Chief of the Laboratory of Experimental and Computational Biology, National Cancer Institute, for making possible the continued development of WinSAAM. We also thank Dr Peter Adamson and Dr Jan Erik Henriksen for stimulating our interest in the problems discussed here.

REFERENCES

1. Rowland M, Tozer TN: Clinical Phamacokinetics, Concepts and Applications. Philadelphia, PA, Lippincott Williams & Wilkins, 1995

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

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

4. Boston RC, Greif PC, Berman M: Conversational SAAM-an interactive program for kinetic analysis of biological systems. Computer Programs Biomed 13:111-119, 1981

5. Barrett PH, Bell BM, Cobelli C, et al: SAAMII: Simulation, Analysis, and Modeling software for tracer and pharmacokinetic studies. Metabolism 47:484-492, 1998

6. Wastney ME, Paterson BH, Linares OA, et al: Investigating Biological Systems Using Modeling. San Diego, CA, Academic Press, 1999

7. Van Milgen J, Boston R, Kohn R, et al: Comparison of available software for dynamic modeling. Ann Zootech 45:257-273, 1996 (suppl 1)

8. Chu SC, Berman M: An experimental method for the solution of systems of ordinary differential equations. Commun Assoc Comput Machinery 17:699-702, 1974

9. Seber GAF, Wilde CJ: Nonlinear Regression. New York, NY, Wiley, 1989

10. Nelder JA, Mead R: A simplex method for function minimization. Comput J 7:308-313, 1965

11. Wagner JG, Wilkinson PK, Ganes DA: Parameters Vm and Km for elimination of alcohol in young male subjects following low doses of alcohol. Alcohol Alcohol 24:555-564, 1989

12. Finegood DT, Pacini G, Bergman RN: The insulin sensitivity index: correlation in dogs between values determined from the intravenous glucose tolerance test and the euglycemic clamp. Diabetes 33:362-368, 1984

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

14. Caumo A, Cobelli C: Hepatic glucose production during labeled IVGTT: Estimation by deconvolution with a new minimal model. Am J Physiol 264:E829-E841, 1993

15. Caumo A, Vicini P, Zachwieja JJ, et al: Undermodeling affects minimal model indexes: Insights from a two compartment model. Am J Physiol 276:E1171-E1193, 1999

16. Lyne A, Boston R, Pettigrew K, et al: EMSA: A SAAM service for the estimation of population parameters based on model fits to identically replicated experiments. Comput Meth Programs Biomed 38:117-151, 1992

17. Boston R, Lyne A, Mc Nabb T, et al: Kinetic models to describe populations: A strategy for summarizing the results of multiple studies, in Siva Subramanian KN, Wastney ME (eds): Kinetic Models of Trace Element and Mineral Metabolism During Development. Boca Raton, FL, CRC Press, 1995, pp 359-372

18. Lyne A, Boston R: Documentation of SAAM's Multiple Studies Capability. NIH Contract Report Contract # 263-87-C-0072