The Analyst

AGANTG: a Microsoft EXCEL 5.0–Visual Basic routine for the analysis of dose–response data

Peter J. Zielinski*[a], and Steve Buckner[b]

[a] *Abbott Laboratories, D9P9, AP32-3, Abbott Park, IL 60064-3500, USA. E-mail: zielinp@megsinet.net*

[b] *Abbott Laboratories, D47C, AP9, Abbott Park, IL 60064-3500, USA*

A Microsoft EXCEL 5.0 program was developed to evaluate data from biochemical and functional bioassays, an important step in drug discovery. The program accommodates both agonist and antagonist data. The program, written entirely in Visual Basic, is compatible with both Macintosh and PC platforms. Data are conveniently entered into a worksheet following only a few simple rules. The program performs complex data analysis and outputs calculated and graphic results to EXCEL worksheets. A set-up routine with a convenient dialog box offers the user controls regarding data analysis and results formats. After determining if the data are from an agonist or antagonist assay, the program automatically performs the analysis and outputs results in the proper format. Calculations support Schild analysis for antagonists. An agonist and antagonist were analyzed to illustrate program usage and results generated by the analsis. EXCEL–Visual Basic is a useful and convenient tool for evaluating bioassay data. Data entry is greatly simplified and custom reports can be generated with relative ease. Data are stored in a format that allows for easy editing re-analysis.

Keywords: *Dose–response data analysis; AGANTG; Microsoft EXCEL 5.0; Visual Basic; bioassay*

Rapidly improving hardware and software for desktop computers continue to open doors to what was once the exclusive domain of large computers. Transferring applications to smaller platforms has obvious benefits for the user in terms of cost and convenience, provided that performance is acceptable. Competition among desktop software manufacturers also provides choices to the user. In particular, a number of spreadsheet programs share a basic look, feel and wide range of functionality so that switching between spreadsheet products entails relatively little difficulty.

To be sure, significant differences in capabilities exist between software products, but familiar spreadsheet interfaces flatten the so-called learning curve for users. Consequently, software developers who take advantage of spreadsheet interfaces expect a higher level of acceptance from users. However, developers must also be able to perform complex and diverse tasks in their applications. For the developer of such applications, a product that combines the spreadsheet user interface with a powerful programming language is highly desirable.

One popular product that has successfully combined these features is Microsoft EXCEL 5.0. This version of EXCEL has inherited the many advantages of EXCEL 4.0 cited by others,[1] especially the familiar feel, ease of use, flexibility and wide range of features. Moreover, EXCEL 5.0 combines extensive spreadsheet functionality with a rich programming language, Visual Basic (VB). This superior combination is useful for many applications and is well suited for the important tasks of evaluating dose–response drug binding and *in vitro* functional data.

Briefly, many drugs work in a lock and key fashion. When tissue containing receptors, the locks, are exposed to a solution containing active drug molecules, the keys, the receptors typically change their shape upon binding with the drug. This change causes a chain of other physiological changes to occur, determining what effects the drug ultimately have. Drugs are categorized as either agonists or antagonists. When an agonist binds, it initiates some physiological reaction. When antagonists bind, they block the effects of agonists, thereby reducing their activity.

A common test for agonist activity is to immerse receptor-bearing tissue in a solution containing some known concentration of the drug and then to measure how the shape of the tissue changes, for example, by monitoring changes in tension that a tissue suspended between two wires exerts before and after addition of the drug. The change in tension is a direct measure of drug activity. A more complicated scheme is required for antagonists, but the essential relationship between drug activity and observed response remains. Here, the tissue is exposed to solutions having various concentrations of a reference agonist besides the antagonist being tested. The activity of the antagonist is determined by how well it prevents the agonist from performing its function. For example, a more active antagonist may mean that more agonist is required to achieve a similar tension. By repeating this measurement for a series of drug concentrations, a dose–response curve may be constructed and the activity of either an agonist or antagonist characterized by mathematically analyzing the curve.[2–4]

An EXCEL 5.0 VB computer program for analyzing dose–response data is described in this paper. EXCEL 5.0 worksheets are used for data input and the output of results. Data are input through a worksheet following a few simple rules. A dialog box provides control over data analysis and reports. The dialog box contains check boxes and entry fields that may be modified at any time before or after data analysis. In this way, the same data can be analyzed in different ways. Computed results and graphs are produced from the data through VB modules that run on either PC and Macintosh platforms.

Program usage

Starting the program and entering data

The program is organized into a number of subroutines, the main routines being AGANTG (AGonist–ANTaGonist analysis) and AGANTG_SetUp. Table 1 gives a short description of the most important of these. AGANTG and AGANTG_SetUp are accessed by the user as EXCEL add-ins under the Tools menu. Like any other add-in, these add-ins may be configured to load automatically upon starting EXCEL or may be loaded at some later time using the Tools menu.

When the user chooses AGANTG_SetUp from the Tools menu, the Data Input worksheet of the currently open workbook is automatically formatted for data entry according to previously stored parameters. Subsequently, the dialog box illustrated in Fig. 1 displays a series of choices to the user. With

Table 1 Important routines and their description

Routine	Description
AGANTG	Main routine for data analysis stage. Directly or indirectly calls nearly all subroutines
GET_XDATA	Retrieves concentrations from Input Data and puts into an array for processing
GET_YDATA	Retrieves responses from Input Data and puts into an array for processing
Probit	Performs probit analysis
TransGraphData	Collates data for graphing and statistical analysis. Calls many subroutines to complete its 'messy' tasks
AGANTG_SetUp	Main routine for set-up stage. Displays the Input Dialog box and stores set-up parameters
CallMrq	Entry point for non-linear analysis. Calls several numerical routines
Estimated_C	Performs initial estimates for *b* and *c* parameters used in 4PLC routine

the dialog box, the user may override previous format parameters and save new ones. There are currently 12 such parameters.

Dialog box entries 'Last Input Row' and 'Last Input Column' define the range of worksheet cells containing input data. In Fig. 1, these entries are 20 and 10, respectively. The program color highlights these cells on the Data Input worksheet, illustrated in Fig. 2, to indicate where data to be processed are located. Data outside this region and empty columns within the region are ignored. The field labeled 'Maximum Iterations' limits attempts by the non-linear curve fitting routine in refining parameter estimates of dose–response curves. Another check box directs the AGANTG routine to perform pA_2 calculations if the data set is an antagonist run or to perform pD_2 calculations in the case of agonist data. The user is not required to specify whether the data are of type agonist or antagonist as the program is able to distinguish these data types, as will be explained below. The user may choose that standard errors rather than standard deviations be displayed on reports and chart error bars. Charts are generated by checking the appropriate boxes. Another check box permits the automatic printing of results. Finally, one box labeled 'CHOOSE AGANTG DEFAULT SETTINGS' will automatically set default choices for an AGANTG analysis. Specific settings, such as the 'Last Input Column' can be overridden by clearing the AGANTG default settings box and entering the desired number of columns. Numerical entries are made with either spinner buttons or by direct entry into the appropriate field. In this way, the AGANTG routine may be configured for purposes other than dose–response curve analysis. For example, it may be used as a general non-linear curve fitting routine.

Data may be entered on to the first worksheet either before or after invoking the SetUp routine. The SetUp routine does not erase this data and may be called from the Tools menu. The role of the SetUp routine is to highlight input data and tell AGANTG what tasks to perform. However, AGANTG expects the data to conform to a few simple rules. While explaining these rules, it will be helpful to refer to Figs. 2 and 3.

The top row of the Data Input sheet is reserved for what shall be referred to as 'titles'. A title identifies information stored in the cells below it, in the same column within the data region selected. The leftmost column of data is assumed to contain values of the independent, or *x*, variable. If we are analyzing dose–response curves, the leftmost column would typically contain agonist concentrations, as illustrated in Fig. 2. The title of this first column, shown as [ ] in this example, is optional and has no effect on how the data are analyzed. The titles of the remaining columns with the data input region do have an impact on how data are analyzed, as explained below. The data themselves need not be in ascending or descending order as they will be sorted by the program. Empty rows or columns within the data entry region are ignored.

The next non-empty column of data is assumed to be that of a vehicle, a control containing only an agonist. Data are

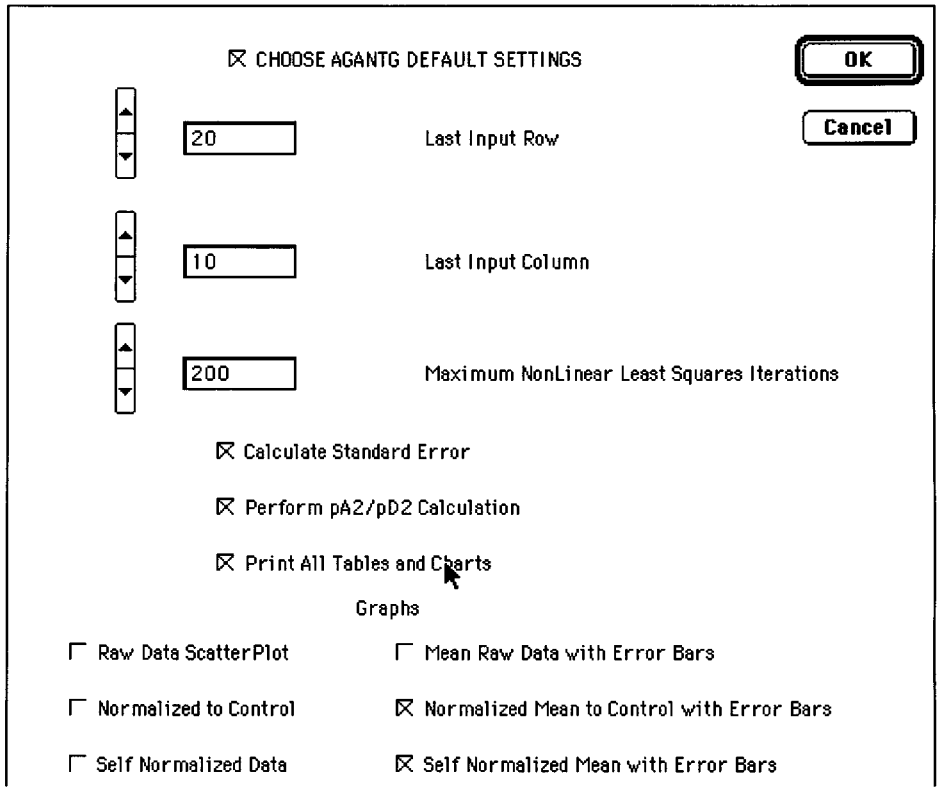


Fig. 1

identified as a control when its title consists of a single string of alphanumeric characters, such as 'veh' in Fig. 2. The title of the next column, 'ant 123 3e − 9', has two parts to it. The first part, 'ant 123' , is an antagonist identifier. The program knows that this sample is an antagonist because the title has a second, numerical part, '3e − 9'. If an antagonist is found in any title

antagonist

	A	B	C	D	E	F
1	[pe]	veh	ant123 3e - g	veh	ant 123 1e - 8	veh
2	1.00E-08	49	45	51	50	49
3	3.00E-08	49	45	51	50	49
4	1.00E-07	56	45	75	50	50
5	3.00E-07	80	55	125	51	60
6	1.00E-06	231	85	238	75	150
7	3.00E-06	356	200	400	135	238
8	1.00E-05	438	338	475	313	313
9	3.00E-05	463	388	475	388	313
10	1.00E-04	463	413		425	
11	3.00E-04		413		425	
12	1.00E-03					
13	3.00E-03					
14	1.00E-02					
15	3.00E-02					
16						
17						
18						
19						
20						
21						
22	First/Last Initials	sab				
23	Ref Agonist:	A157403.3				
24	Tissue:	DOG PROSTATE				
25	Agonist Test? (Y/	n				
26	Sample Name:	ant123				
27	Test Date:	Apr 5 1996				
28	Report Time:	2:24 PM				
29	Report Date:	May 23 1996				
30	Lot Number:					

Fig. 2

Agonist Example

	A	B	C	D	E	F	G	H	
1	[]	ref	sample	ref	sample	ref	sample	ref	sar
2	1.00E-10	55	48	49	46	49	50	48	
3	3.00E-10	55	48	49	46	49	50	48	
4	1.00E-09	55	49	49	47	49	51	48	
5	3.00E-09	55	155	49	78	49	128	48	
6	1.00E-08	55	180	49	120	49	150	48	
7	3.00E-08	58	180	49	135	49.1	190	48	
8	1.00E-07	59		53	135	55	190	63	
9	3.00E-07	83		83		75		100	
10	1.00E-06	120		125		110		125	
11	3.00E-06	175		175		165		145	
12	1.00E-05	175		175		165		145	
13	3.00E-05								
14									
15									
16									
17									
18									
19									
20									
21									
22	First/Last Initials	SB							
23	Ref Agonist:	PE							
24	Tissue:	A7313							
25	Agonist Test? (Y/	Y							
26	Sample Name:	A4473							
27	Test Date:	33742							
28	Report Time:	11:05 AM							
29	Report Date:	May 22 1996							
30	Lot Number:	L#3434							

Fig. 3

within the data input region, the program assumes that the data represent an antagonist experiment. Otherwise, an agonist experiment is assumed. Fig. 2 shows the data set for an antagonist and Fig. 3 illustrates an agonist data set.

Information to be processed may be entered into one other area besides the data input region. The area is in the second column just below the data input area, as illustrated in Fig. 3. Headings in the first column indicate information that may be entered to aid in assay identification. Entry of report time and date information by the analyst is not required as these cells are filled automatically. Agonist, tissue, sample and date information are copied on to all charts by the program. The Agonist Test cell, which is next to that labeled 'Agonist Test?', is filled by the analyst to indicate whether or not the assay is a test for an agonist. This cell is used later as a check to help determine if a mistake has been made entering column titles. All of these assay identification fields and headings are copied to all worksheets containing final results. An illustration of one of the graphs output for a typical antagonist assay is given in Fig. 4.

Treatment of data

Data validation and analysis

When a workbook is opened and the AGANTG routine started by selecting RUN AGANTG from the Tools menu, the program immediately erases any previously existing charts and worksheets in the workbook, except for the Data Input worksheet. The program then tests the data for errors. The first column in the data region of Data Input, the independent *x* values, must contain at least two distinct values for any analysis to continue. Titles without correct formats are highlighted and the program is halted. The same is true if characters are accidentally placed into regions reserved for numeric data. Additional data checks insure that the response data is valid. For example, any response column (the dependent *y* values) not associated with at least two distinct agonist concentrations, or not varying with the *x* values, will be ignored after appropriate alerts are issued since further calculations on these data are pointless. In any case, a cleaned up version of the data is stored on the worksheet 'Processed Data' and, if it was found acceptable, used for all future calculations. Regression is then attempted, as described next.

When a *y* column is associated with at least four distinct *x* values, the *x*–*y* data are passed to a non-linear regression routine that finds parameters *a*, *b*, *c* and *d* according to the four parameter logistic model (4PLC):

$$y = d + (a - d)/[1 + (x/c)^b] \quad (1)$$

where *y* represents the observed response and *x* its associated dose. Parameter *a* is the response at $x = 0$, *b* a 'slope factor' describing the steepness of the curve, *c* the EC_{50} and *d* the response for infinite dose. For a drug, the EC_{50} corresponds to the concentration of the drug necessary to produce 50% of the maximum possible response. Eqn. (1) is a non-linear model equation that describes sigmoidal dose–response curve data.[4–6] To find the four parameters for a particular dose–response curve, the response values in a given *y* column are regressed against the dose values in the *x* column using eqn. (1).

Non-linear regression requires initial estimates of the four parameters described above. AGANTG automatically performs these estimations without user intervention and then attempts regression on each valid *x*–*y* data set using eqn. (1). Our method of choice for this purpose is the Levenberg–Marquardt algorithm.[7,8] Once the calculation is complete, the parameters returned are tested for reasonableness. If the parameters fail any

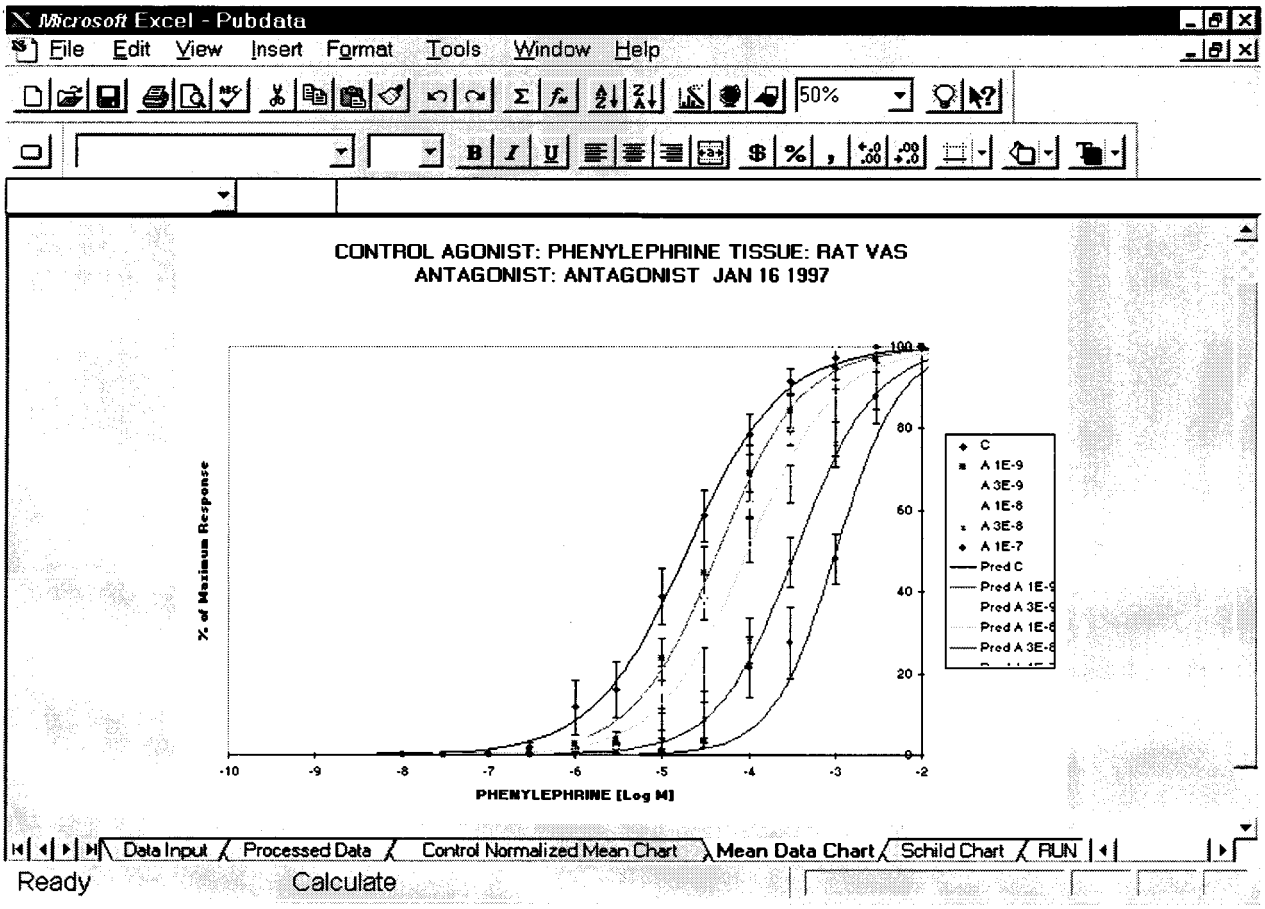


Fig. 4

of these tests, a linear regression is attempted using the probit model eqn. (2) rather than the 4PLC model:

$$G^{-1}(y/y_{max}) = m + \beta \log x \quad (2)$$

where G^{-1} is the inverse Gaussian cumulative probability function. For a given probability value p in the interval [0,1], $G^{-1}(p)$ returns the value along the x-axis that yields that probability. Thus, $G^{-1}(0) = -\infty$, $G^{1}(0.5) = 0$ and $G^{-1}(1) = \infty$. Once regression is complete, the EC_{50} is given by

$$EC_{50} = 10^{-m/\beta} \quad (3)$$

A particular dose–response curve is abandoned if both the 4PLC and probit regression are unsuccessful. The parameters found, predicted y values and residues are stored in a worksheet labeled 'Results' and may be viewed, if desired. Parameter values are later retrieved from this worksheet for further calculations and reports.

When all regressions have been completed and the results stored, the program determines whether the data represent an agonist or antagonist assay by checking for the presence of a second numerical component in the title, as described above. If this determination contradicts what is entered by the Agonist Test cell, a warning is issued and the analyst given an opportunity to abort the analysis. If the analysis is not aborted, the entry in the Agonist Test cell is ignored. Subsequent calculations and reports depend on the nature of the assay as decided based upon data titles.

For antagonist, the objective is the calculation of pA_2. According to Schild, 'pA_x is defined as the negative logarithm to base 10 of the molar concentration of an antagonistic drug which will reduce the effect of a multiple dose (x) of an active drug to that of a single dose'. This calculation involves the quantity known as the dose-ratio (DR), given by[3]

$$DR = EC_{50\,Antagonist}/EC_{50\,Control} \quad (4a)$$

where DR is the EC_{50} of the agonist in the presence and absence of antagonist. The quantity $DR - 1$ is given by

$$DR - 1 = EC_{50\,Antagonist}/EC_{50\,Control} - 1 = (EC_{50\,Antagonist} - EC_{50\,Control})/EC_{50\,Control} \quad (4b)$$

The pA_2 is computed by performing a linear regression using the model eqn. (5) and then using the fitted slope, β, and intercept parameter, m, in eqn. (6):

$$\log(DR - 1) = m + \beta \log x \quad (5)$$

$$pA_2 = -m/\beta \quad (6)$$

When the data contain more than one level of antagonist concentration, AGANTG produces a plot of the data transformed according to eqn. (5) together with the fitted line. If only one antagonist concentration is used, a constrained regression is performed in which the slope in eqn. (5) is held to the value 1[9]; m is then found and again used in eqn. (6), but no pA_2 chart is generated. Examples of default chart type that AGANTG generates are illustrated in Fig. 4.

The objective is different for agonist analysis. The first measure of agonist potency is the pD_2 value. This quantity is simply

$$pD_2 = -\log(EC_{50}) \quad (7)$$

For an agonist, the Index is the measure of potency. The Index is defined as the ratio of the effective concentration of the agonist control divided by the effective concentration of the sample agonist:

$$\text{Index} = EC_{50\,\text{control agonist}}/EC_{50\,\text{sample agonist}} \quad (8)$$

A large Index indicates a potent sample agonist relative to the control agonist. Relative efficacy, the percentage of a sample's maximum response relative to that of the control, is reported in the Summary Report as illustrated in Fig. 5. For agonists, key individual results are illustrated in Table 2.

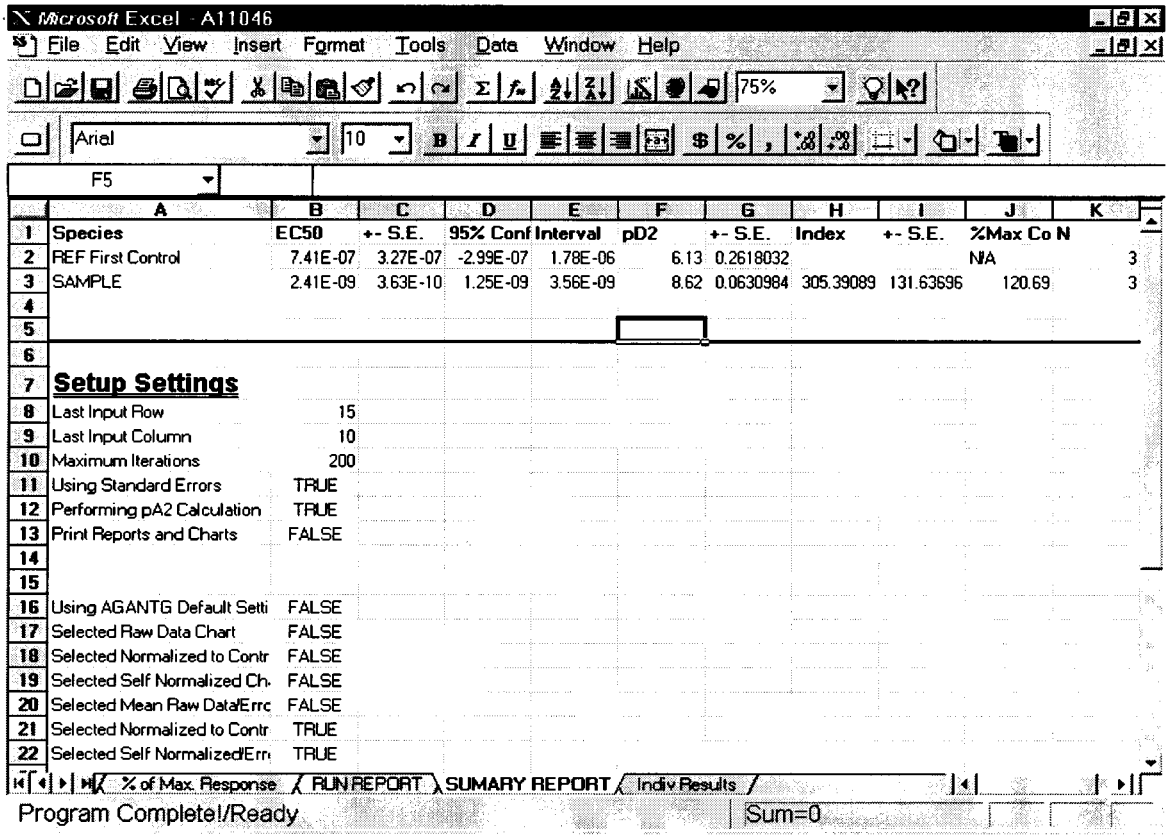
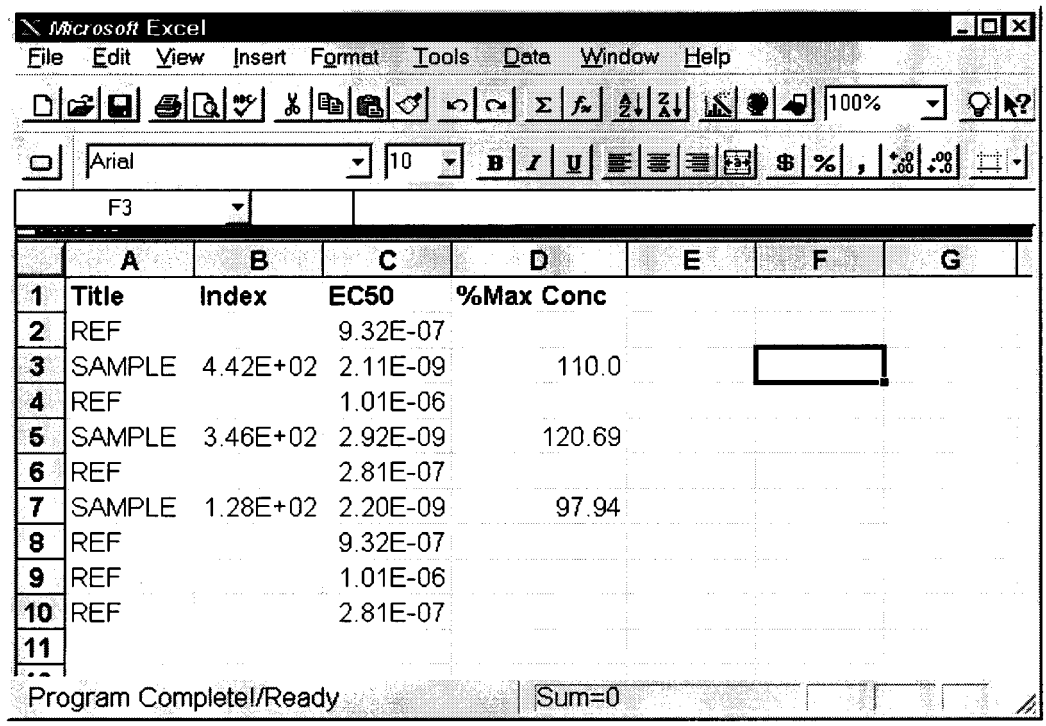


Fig. 5

Table 2 Individual results for agonist run data in Table 3



	A	B	C	D	E	F	G
1	Title	Index	EC50	%Max Conc			
2	REF		9.32E-07				
3	SAMPLE	4.42E+02	2.11E-09	110.0			
4	REF		1.01E-06				
5	SAMPLE	3.46E+02	2.92E-09	120.69			
6	REF		2.81E-07				
7	SAMPLE	1.28E+02	2.20E-09	97.94			
8	REF		9.32E-07				
9	REF		1.01E-06				
10	REF		2.81E-07				
11							

Table 3 Agonist run data

[ ]	REF	SAMPLE	REF	SAMPLE	REF	SAMPLE	REF	SAMPLE
1E − 10	55	48	49	46	49	50	48	45
3E − 10	55	48	49	46	49	50	48	45
1E − 09	55	49	49	47	49	51	48	50
3E − 09	55	155	49	78	49	128	48	118
1E − 08	55	180	49	120	49	150	48	140
3E − 08	58	180	49	135	49	190	48	140
1E − 07	59		53	135	55	190	63	
3E − 07	83		83		75		100	
1E − 06	120		125		110		125	
3E − 06	175		175		165		145	
1E − 05	175		175		165		145	
3E − 05								

Statistics

AGANTG furnishes various report sheets. Of these, the Run and Summary reports include statistics. These two sheets display several types of statistics that are described next.

Consider the agonist run data shown in Table 3 and the Run report given in Table 4. Simple average values are given on the Run report in the columns containing Test Values. These values are the mean response values observed for a given experiment at each of the agonist concentrations used during the experiment. For example, If we look at the columns in Table 3 labeled 'REF' at a concentration of 1E − 05, the average of the values 175, 175, 165 and 145 is given as 165 at that same concentration in Table 4. Values labeled %Resp in Table 4 represent responses after they have been self-normalized and averaged. The self normalization for y_i, the response associated with the ith concentration, is performed using the equation.

$$y_{\text{norm}} = 100(y_i - y_{\text{min}})/(y_{\text{max}} - y_{\text{min}}) \quad (9)$$

where y_{norm} is the self-normalized response value for a particular concentration and y_{min} is the smallest and y_{max} the largest response observed in the experiment in question.

Values in Table 4 under the heading'Norm%Resp' are computed using eqn. (10), which is similar to eqn. (9) except that the maximum and minimum y values in the denominator are those of the control immediately preceding the column of interest. These are labeled y_{cmax} and y_{cmin}:

$$y_{\text{cnorm}} = 100(y_i - y_{\text{min}})/(y_{\text{cmax}} - y_{\text{cmin}}) \quad (10)$$

To obtain an idea of how values fluctuate around raw or normalized averages, a column labeled '+ − SEM' or '+ − SD' is provided, depending upon whether the standard error of the mean or standard deviation was chosen during the SetUp dialog. These values are computed using either eqn. (11) or eqn. (12), with n being the number of data points. Table 4 indicates that the standard error was chosen for the illustrated analysis.

$$\text{SD} = \sqrt{\Sigma(y_i - y_{\text{iave}})^{2/}(n-1)} \quad (11)$$

$$\text{SEM} = \sqrt{\Sigma(y_i - y_{\text{iave}})^{2/}n(n-1)} \quad (12)$$

Eqns. (11) and (12) are also used in the beginning section of the Summary report sheet to compute results labeled '+ − SEM' or

'+ − SD'. However, the SE values of the pA_2 results, in the pA_2 section, are computed differently, as explained by eqn. (5.4) of Tallarida and Murray.[10] Likewise, the SE of the slope is computed using eqn. (5.2) of Tallarida and Murray.[10.]

Values under the headings EC_{50} and so on refer to average values having the number of points listed under the heading 'N'. In all estimates of confidence intervals, limits are obtained from eqn. (5.3) of Tallarida and Murray.[10] All intervals in the Summary report are estimated at 95% confidence. The coefficient of linear correlation, *r*, follows a standard equation:[11]

$$r = \Sigma[(x_i - x_{ave})(y_i - y_{ave})]/\sqrt{\Sigma(y_i - y_{ave})^2 \xi(x_i - x_{ave})^2} \quad (13)$$

Program validation

The ALLFIT program is the reference of choice for validating new programs. Table 5 summarizes final results produced by the AGANTG and ALLFIT programs using the same data. Intermediate results, including some of the fit parameters, are not reported by ALLFIT and are not compared here. Table 5 contains data for both agonists and antagonists. Each agonist experiment was repeated four times and each antagonist experiment was repeated six or eight times for the purpose of this comparison. The data were then analyzed by both programs using a maximum of 200 iterations for the curve fitting step.

The two methods agree well, as illustrated by a comparison of two of the most important results contained in Table 5. Linear regression of equally weighted AGANTG EC_{50} results for agonists against the ALLFIT results produces a line with a slope of approximately 0.995, an intercept of approximately 0.000 000 7 and a correlation of 1.000 00. A similar analysis of antagonist pA2 results produce a slope of approximately 1.001, an intercept of −0.026 and a correlation of 0.999.

The small differences observed between the results of these programs are most likely due to the non-linear regression step required for analysis. Because non-linear regression is more complicated than ordinary linear regression, some differences are expected. The many implementation details of non-linear algorithms may produce differences in results even when using the same method, such as the Levenberg–Marquardt algorithm. Some likely sources of the observed differences in results may include differences in starting parameter estimates and differences in the implementation of regression algorithms themselves.

Conclusions

EXCEL 5.0 combines a user interface and programming language that benefits both software users and developers. EXCEL–VB is a powerful tool for developing complex applications such as AGANTG. Graphical objects, worksheets, charts, menus and dialog boxes can be easily created and manipulated by the developer through VB. VB is a structured language with an extensive library of classes and functions to aid developers in their task.

Users benefit because applications developed with EXCEL have the familiar look and feel of spreadsheets. Users face less needless anxiety since they can concentrate on completing their work instead of learning software tools. Best of all, the user obtains all the functionality of EXCEL on top of the particular application. In the present example, this means that users can cut, paste, import or export data and objects to and from other spreadsheets and applications. Such flexibility can greatly reduce tasks such as entering data, writing reports and transferring results to a database.

The AGANTG program is an important application that illustrates the power of the EXCEL–VB combination. Armed with a few simple rules, the user may enter agonist or antagonist experimental data into a worksheet and perform complex analysis with the click of a mouse. Data or analysis options can be edited and the data re-analyzed. One may, for example, analyze intermediate data then later add to those data and re-analyze.

Analysis options are edited whenever necessary using a convenient dialog box. During analysis, the data are scrutinized and any detected errors highlighted. This step prevents wasting time spent looking for subtle mistakes. Additional messages and message boxes are used to alert the user to the status of the analysis and to issue warnings when appropriate.

Table 4 Run report for data in Table 3

[Agonist]	REF Test Val	+ − S.E.M.	REF %Resp	+ − S.E.M.	Norm %Resp	+ − S.E.M.
1.00e − 10	50.3	1.6	0.0	0.0		
3.00e − 10	50.3	1.6	0.0	0.0		
1.00e − 09	50.3	1.6	0.0	0.0		
3.00e − 09	50.3	1.6	0.0	0.0		
1.00e − 08	50.3	1.6	0.0	0.0		
3.00e − 08	51.0	2.3	0.6	0.6		
1.00e − 07	57.5	2.2	6.8	2.9		
3.00e − 07	85.3	5.3	31.6	7.4		
1.00e − 06	120.0	3.5	61.6	6.2		
3.00e − 06	165.0	7.1	100.0	0.0		
1.00e − 05	165.0	7.1	100.0	0.0		
3.00e − 05						
[Agonist]	SAMPLE Test	+ − S.E.M.	SAMPLE %Resp	+ − S.E.M.	Norm %Resp	+ − S.E.M.
1.00e − 10	47.3	1.1	0.0	0.0	0.0	0.0
3.00e − 10	47.3	1.1	0.0	0.0	0.0	0.0
1.00e − 09	49.3	0.9	2.0	1.1	1.9	1.1
3.00e − 09	119.8	16.0	62.4	10.4	64.3	13.7
1.00e − 08	147.5	12.5	88.6	7.0	88.2	11.0
3.00e − 08	161.3	13.9	100.0	0.0	99.8	10.8
1.00e − 07	162.5	27.5	100.0	0.0	95.7	25.0
3.00e − 07						
1.00e − 06						
3.00e − 06						
1.00e − 05						
3.00e − 05						

Table 5 Comparison of ALLFIT and AGANTG results

Runs	Model	EC_{50}	SEM/EC_{50}	Lower 95%	Upper 95%	% Max.	Method
4	VAS ala − 7	2.09e − 6	1.56e − 7	1.84e − 6	2.37e − 6	89.1	ALLFIT
		1.98e − 6	2.68e − 7	1.24e − 6	2.73e − 6	89.1	AGANTG
4	SPLEEN alb − 8	2.11e − 4	2.80e − 5	1.70e − 4	2.62e − 4	77.4	ALLFIT
		2.10e − 4	3.01e − 5	1.27e − 4	2.94e − 4	77.4	AGANTG
4	VAS ala − 7	9.20e − 8	1.35e − 8	7.45e − 8	1.14e − 7	92.7	ALLFIT
		9.36e − 8	1.44e − 8	5.37e − 8	1.34e − 7	92.7	AGANTG
4	PROST ala − 17	1.22e − 7	5.04e − 8	7.12e − 8	2.08e − 7	126.7	ALLFIT
		1.20e − 7	5.18e − 8	2.34e − 8	2.64e − 7	126.7	AGANTG
4	AORTA ald − 10	1.70e − 6	5.70e − 7	9.83e − 7	2.93e − 6	62.3	ALLFIT
		1.82e − 6	6.45e − 7	3.18e − 8	3.61e − 6	62.3	AGANTG

Runs	Model	pA2	SEM/pA2 SEM/slope	Correlation *r*	Slope	Low/up 95%	Method
6	PROST 71	6.72	0.16 0.09	0.98	1.03	6.30/7.14	ALLFIT
		6.67	0.15 0.06	0.99	1.04	6.38/6.96	AGANTG
6	URETHRA	5.57	0.30 0.24	0.91	1.05	4.74/6.41	ALLFIT
		5.56	0.13 0.14	0.03	1.12	5.20/5.92	AGANTG
6	AORTA	8.22	0.24 0.11	0.98	1.18	7.57/8.88	ALLFIT
		8.13	0.67 0.17	0.96	1.18	6.26/9.99	AGANTG
8	VAS	8.47	0.24 0.14	0.96	1.22	7.89/9.05	ALLFIT
		8.59	0.36 0.15	0.92	1.16	7.70/9.47	AGANTG
8	AORTA	9.46	0.19 0.05	0.99	1.06	9.00/9.92	ALLFIT
		9.40	0.22 0.05	0.99	1.09	8.87/9.93	AGANTG

The AGANTG program was written in a manner that minimizes the number of inputs required of the user. For example, the program is able to decide if the experiment is that of an agonist or an antagonist from the format of column data headers and perform the correct analysis. In the case of antagonists, the program automatically determines if the pA_2 uses multiple concentrations of antagonists and whether or not a pA_2 plot should be generated. Although this may seem a trivial task, such a hands off approach can save the user both time and aggravation when other seemingly trivial tasks are likewise handled. A few other examples of simple but valuable automated tasks performed by AGANTG that eliminate user inputs include avoiding the entering of the number of data points used in an experiment, the sorting of concentration–response data into the proper order and deciding if data are inhibitory or excitatory.

The authors express their thanks to Guennadi Safronov for his help in debugging the AGANTG program.

References

1 Tack, G., Roselli, K. A., Overhoser, A., and Harris, T. R., *Comput. Biomed. Res.*, 1995, **28**, 24.
2 Arunlakshana, O., and Schild, H. O., *Br. J. Pharmacol.*, 1959, **14**, 48.
3 Waud, D. R., and Parker, R. B., *J. Pharmacol. Exp. Ther.*, 1971, **177**.
4 Rodbard, D., and McClean, S. W., *Clin. Chem.*, 1977, **23**, 112.
5 Rodbard, D., *Clin. Chem.*, 1974, **20**, 1255.
6 Delean, A., Munson, P. J., and Rodbard, D., *Am. J. Physiol.*, 1978, **4**, e97.
7 Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T., *Numerical Recipes, The Art of Scientific Computing*, Cambridge University Press, Cambridge, 1986.
8 Dahlquist, G., and Bjorck, A., *Numerical Methods*, Prentice-Hall, Englewood Cliffs, NJ, 1974.
9 Furchgott, R. R., in *Handbook of Experimental Pharmacology*, ed. Blaschko, H., and Muscholl, E., Springer, Berlin, 1972, vol. 33, pp. 283–335.
10 Tallarida, R. J., and Murray, R. B., *Manual of Pharmacologic Calculation with Computer Programs*, Springer, New York, 2nd edn., 1987.
11 Snedecor, G. W., and Cochran, W. G., *Statistical Methods*, Iowa State University Press, Ames, IA, 7th edn.,1980.

Paper 8/04438D
Accepted June 12, 1998