

Distance Tutoring in Engineering Economics: Equivalence Modeling

John H. Ristroph, Ph.D., P.E.
University of Louisiana at Lafayette

Abstract

This paper focuses on the creative elements of designing a system suitable for tutoring distance students in equivalence modeling. The design of Equivalence Tutor seeks to replicate many of the essential elements of a professor's office hours that frequently are unavailable to distance learning students.

Tutor uses Microsoft's Excel as a platform, fully drawing upon its graphical, data management, and mathematical resources. Spreadsheets, per se, are inadequate for tutoring purposes, so Excel's capabilities are extended by using its internal programming language, Visual Basic for Applications.

Introduction

Engineering economics is a unique course that requires students to develop mathematical skills in equivalence modeling, to apply concepts and solve problems using economic criteria, and to master rule and memory oriented topics such as taxes. Students must rely on different learning skills, and this makes teaching the course a fascinating challenge. One common thread to the different components of the course is that it does not require a physical laboratory, and this makes it a candidate for web based teaching.

Computer applications in engineering economics have a long history. Numerous texts have provided software supplements for years, and now posting lecture notes on web sites is not unusual. Terpenney, Sullivan, et al have taken this concept one step further. They reviewed web related learning issues and developed a virtual classroom [1] for engineering economics that even allows for open ended projects with industrial interaction [2]. Their web site [3] allows students to interactively select among a variety of topics and even take multiple choice quizzes. Prompts congratulate correct answers and provide general guidance for wrong answers.

The current project restricts its focus to equivalence modeling to allow interaction at an equation level much as would occur in a professor's office. The following sections discuss basic design considerations and then examine subsystems and their features in more detail.

Basic Design Considerations

At one end of the design spectrum is a system that swiftly and accurately performs any calculation requested by the student, but does not offer any guidance at all. This allows a student to explore and learn by doing, but does such a system really offer tutoring or is it merely an advanced calculator? At the other end of the spectrum is a system that offers students a series of structured choices and prompts, refusing to proceed until the desired answer is given. Such a system can aid the learning of mechanics, but to what extent does it encourage independent thinking?

Tutor is a hybrid of the foregoing extremes. The student must take the active step of entering his or her own equations. If an equation contains mistakes to the point that it cannot be implemented, then the system offers guidance, but not necessarily a specific solution. The student must keep trying until an executable equation is entered. Even then, it is possible that the equation will not decompose the original problem in a way that moves the student closer to a solution. The system displays cash flow diagrams to help students understand the effects of each equation. When appropriate, the system issues warning, and it allows undoing steps, but the student must make his or her own decisions.

The computational and graphical requirements would be excessive unless the primary system is downloaded only once. For example, the nonlinear scaling algorithm that prevents overly small or large arrows on diagrams is one of many features that consume resources in a manner transparent to the user. The Excel spreadsheet is roughly four megabytes, and it can be compressed (zipped) to approximately one-half megabyte. Any other files needed for downloading problems or reporting homework activity are much smaller and can be transmitted as needed.

System Overview

Figure 1 shows that Tutor consists has three windows corresponding to its basic functions: graphics (Diagram), data management (Data), and user interaction (Main). The display shows the screen after the Problems button has been clicked and a problem selected. The user also has clicked the downward arrow in the Diagram window. This selects it and shows its name and amount in the name box on the upper left of the display: B5=-300.00. The name of a cash flow or equivalent is its storage location on sheet Amounts of the Data window.

Figure 2 shows the second sheet of the Data window that contains interest values. Clicking on a column heading in sheet Interest such as (Stated) Interest or in the Main window causes the Office Assistant to explain that column. Notice that the last statement of the current help message indicates that the interest entries are named. Clicking on the cell containing 10.00% causes its name (either i or iA) to be displayed. Names of cells are important because Tutor maintains accuracy using names in specifying equivalence operations instead of allowing possibly rounded or mistyped numeric entries.

Clicking the Question button produces a message box giving the problem statement, to determine the equivalent at time 0 in this example. The first step in solving this problem is evaluating the equivalent of B5 at time 0:

$$E_{0 \text{ of } B5} = -300.00(P|F, i, 5-0) \quad (1)$$

The last parameter of the factor is expressed as the difference of two numbers. This frequently is done implicitly and only the result is noted, 5 in this instance. However, explicitly showing the subtraction operation helps prevent careless errors and makes it much easier to follow logic, so Tutor requires users to do this explicitly, referring to the last parameter as s-r.

Figure 3 displays the Data window after entering equation (1), but before pressing the Compute button. The columns of the Main window show from left to right that the user:

Place @ t = : places at time 0

Calc(ulation): an equivalent

Amo / Cols: formed by multiplying the amount of flow B5

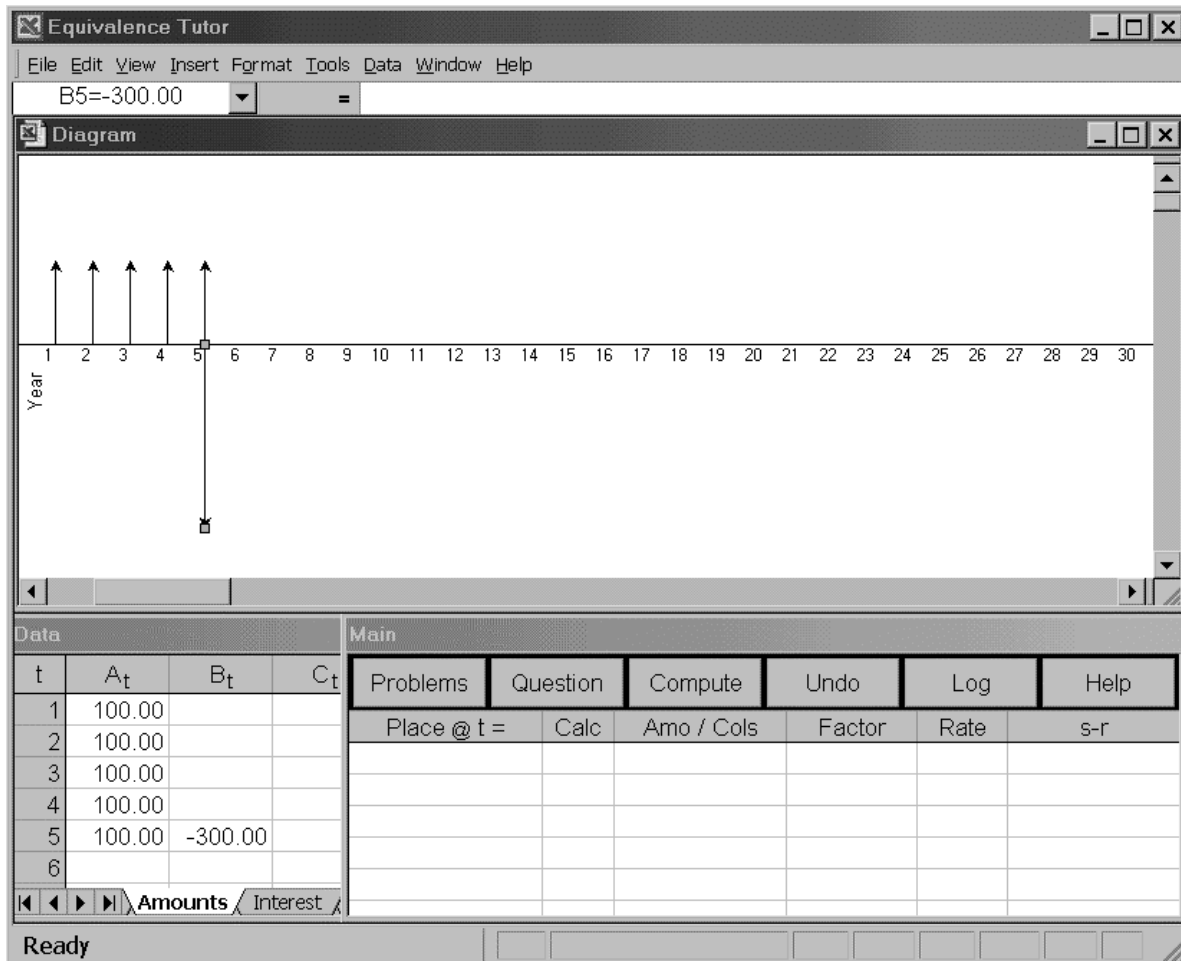


Figure 1. Initial Display

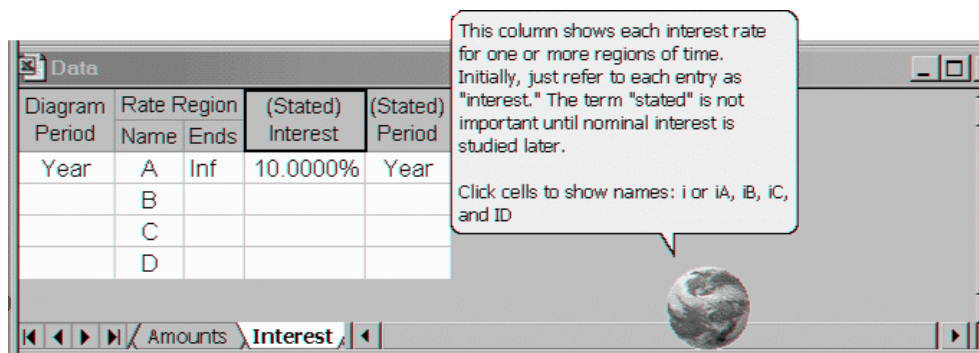


Figure 2. Sheet Interest and the Office Assistant

Factor: by the P|F factor

Rate: using interest rate i (declared in sheet Interest as 10%)

s-r: for 5-0 periods.

Problems	Question	Compute	Undo	Log	Help
Place @ t =	Calc	Amo / Cols	Factor	Rate	s-r
0	Eqv	B5	P F	i	5-0

Figure 3. Entry of First Equation, Before Pressing Compute

Figure 4 shows the system after pressing the Compute button to:

1. check for possible errors,
2. evaluate equivalents,
3. subtract the amounts affected by the equivalence operation,
4. add the equivalent(s),
5. redraw the diagram with equivalents shown as dashed arrows in the same direction as the

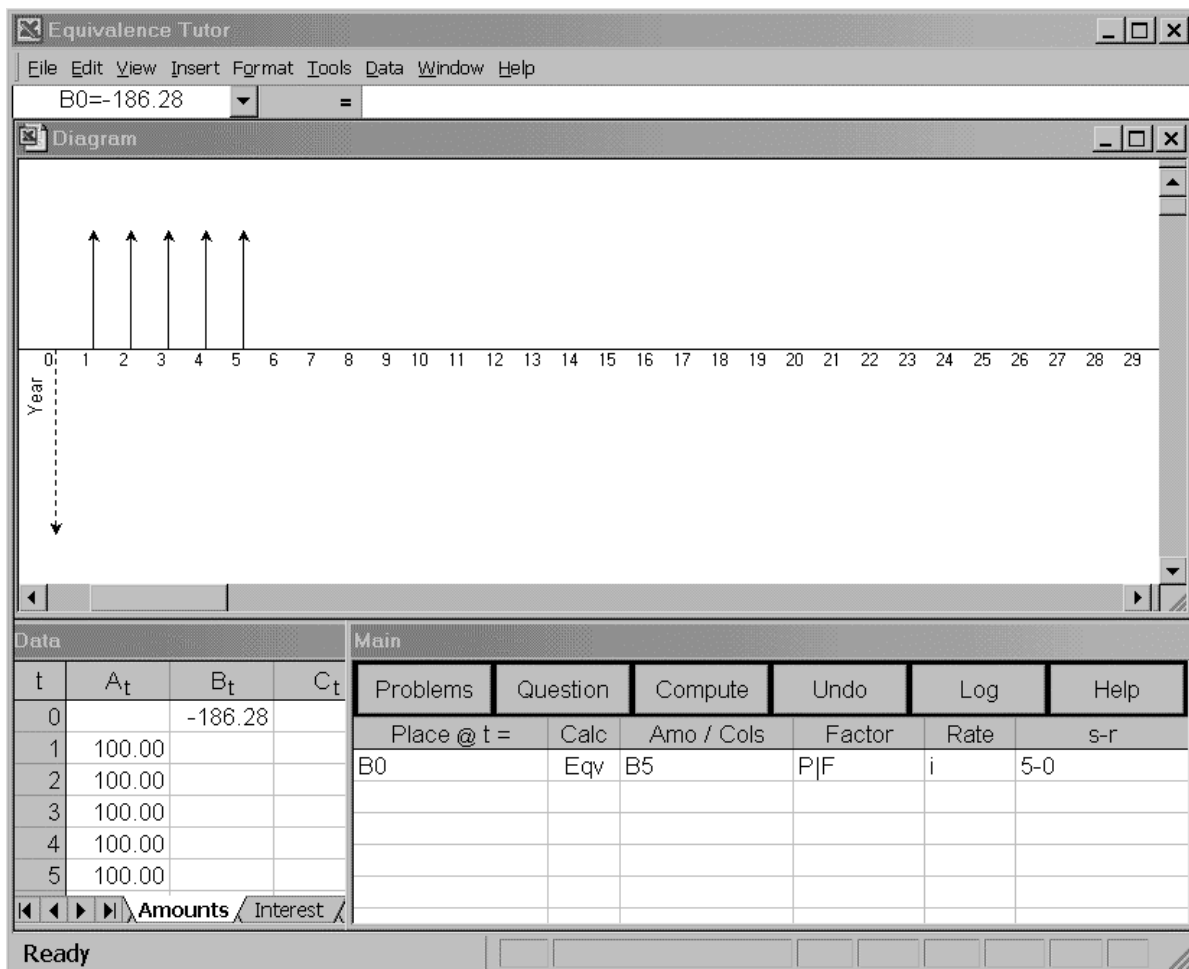


Figure 4. First Equation, After Pressing Compute

replaced flows, and

6. assign names and storage locations.

Notice that Tutor changes the user's initial entry of time 0 in the @ (i.e., Place @ t =) column to the storage location of the equivalent, B0. This informs the user of the names of any new equivalents, thereby facilitating their use in subsequent computations.

Figure 5 displays the system after evaluating the equation:

$$E_{0 \text{ of } A1:A5} = 100.00(P|A, i, 5-0) \quad (2)$$

The last step in solving the problem is to sum the entries at time 0 in columns A and B. The user inputs 0 into the @ column, enters Sum as the calculation, and declares A and B as the columns to be summed (at time 0 only). Figure 6 displays the system after the Compute button is pressed, automatically changing the 0 entry in the @ column to A0.

Tutor uses a coded form of the problem's objective displayed by the Question button to internally solve any problem. Solutions can be a single payment, uniform series, or multiperiod series of equal amounts at time intervals greater than one. User's entering names of values instead of numbers allows Tutor to maintain sufficient accuracy to see if its internal solution matches the result of any equation and to notify the user to log the solution.

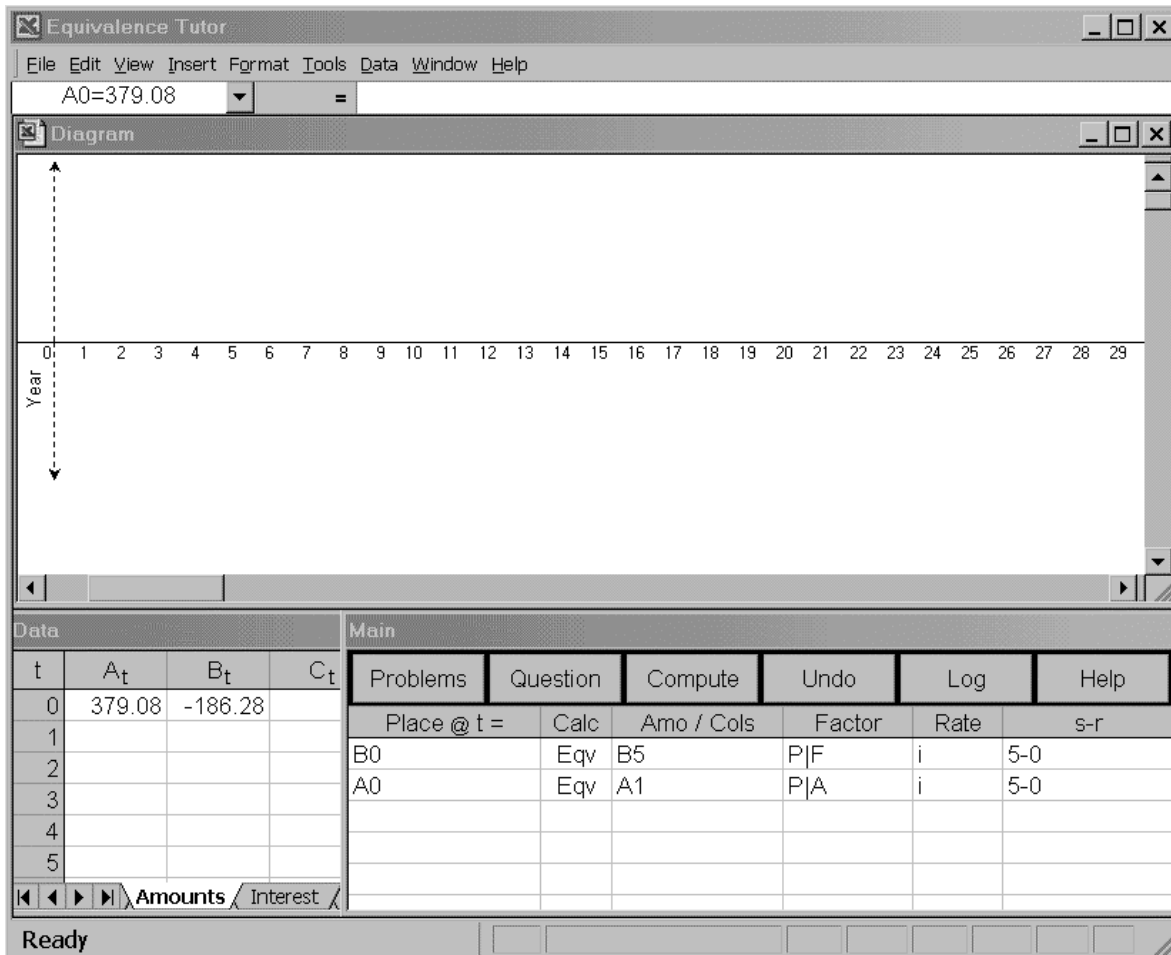


Figure 5. Second Equation, After Pressing Compute

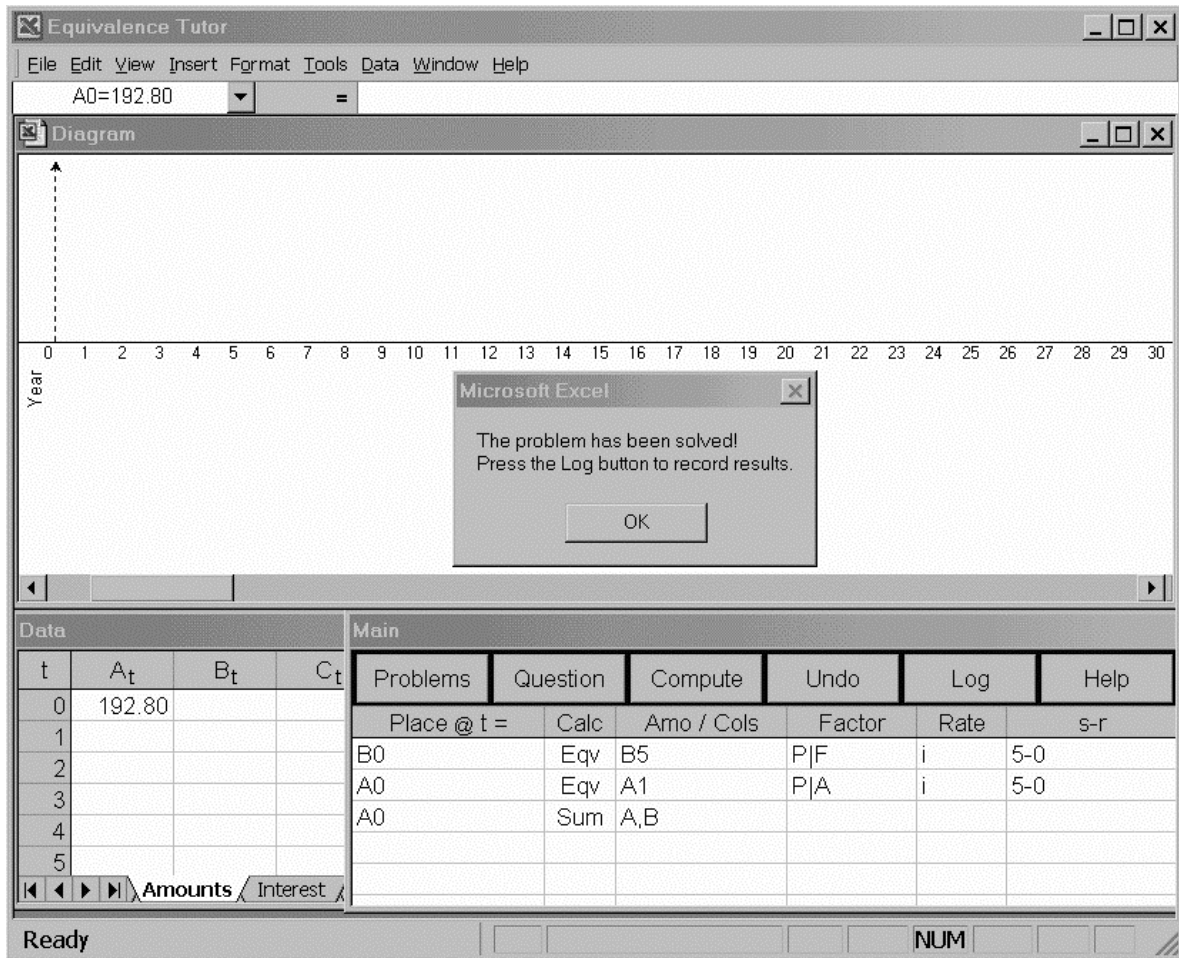


Figure 6. Solution

Storing which problems have been solved, when, and how much time was required allows automated grading and monitoring of students' activities. Storing the log as an encrypted text file prevents tampering, but allows the log to be uploaded easily, and then decrypted and entered into the course database.

Learning by Mistake

If engineering could be mastered simply by attending brilliant lectures, then students would be very happy until receiving their first paycheck. Homework is necessary for students to reinforce and extend classroom presentations. Tutor can measure this activity via its logging system. Moreover, it is designed allow students to make certain types of mistakes and learn from them as efficiently as possible. This section describes how Tutor guides the learning process.

Tutor immediately examines each entry for obvious errors, and notifies the user when one is found. For example, time values must be within system limits, and the fourth parameter must be positive. Also drop-down boxes in the Main window allow selecting the type of computation, factor name, and interest rate name. After **Compute** is clicked, more checks are made for errors that can prevent the computation from taking place. For example, placing the result at time 0 and entering the last factor as 5-1 is an inconsistency that causes an error message and prevents computation.

In general, errors that prevent a correct computation are identified, the user is notified, and corrections must be made. In some cases, notices are displayed for permissible but questionable activities, such as discounting a series when all of the cells of the series do not contain the specified amount. Tutor notes that it must add and subtract cash flows to perform the operation, and then proceeds to do it. Few students need to be taught how to use an Undo button. One can only hope that the minor irritation occasioned by this process provides a stimulus to avoid such mistakes.

Logical errors are far more intriguing, and provide a greater opportunity for learning. For example, consider a uniform series stored in cells A1, A2, ..., A5 and the problem of determining an equivalent series at times 20, 21, ..., 30. Table 1 shows a correct sequence of equivalence equations to solve this problem. The leftmost columns contain first the user's actual entry and then the change made by Tutor after pressing Compute.

Table 1. Correct Equations						
Entry Place @ t =	Change Place @ t =	Calc	Amo / Cols	Factor	Rate	s-r
5	B5	Eqv	A1	F A	i	5-0
19	A19	Eqv	B5	F P	i	19-5
20:30	A20:A30	Eqv	A19	A P	i	30-19

Now consider the not uncommon logical error of putting the single payment compound amount at time 20 instead of 19. Tutor allows entry of the first two equations shown in Table 2 and clicking Compute. Tutor would allow the first two equations, but it would provide an error message for the third one: An s-r entry of 30-20 for A|P forms an equivalent series at times 21, 22, ..., 30 for a flow at time 20, but the @ column requests one at times 20, 21, ..., 30.

Table 2. Logical Error						
Entry Place @ t =	Change Place @ t =	Calc	Amo / Cols	Factor	Rate	s-r
5	B5	Eqv	A1	F A	i	5-0
20	A20	Eqv	B5	F P	i	20-5
20:30		Eqv	A20	A P	i	30-20

At this point, the student might: click Help, view the material on factor use, click Undo, and be successful. Another possibility is changing the s-r entry to 30-21. This produces a notice that amounts of series usually are identified by their first term, but the computation will proceed. The diagram will show the equivalent series at times 21, 22, ..., 30, plus the flow at time 20 that was not included in the equivalence operation. In this case, the diagram quickly shows that something is wrong, but the student must decide what to do next. Tutor cannot guarantee a student's success. It is like a professor who insures that a student's mechanics are correct and explains when they are incorrect, but requires the students to make his or her own decisions.

Models

Tutor supports all of the discrete factors: P|F, F|P, P|A, A|P, F|A, A|F, P|G, A|G, and P|A,g. It also allows the use of single payment continuous compounding factors and represents them as e^+ and e^- . Limitations include an integral time scale and no funds flow modeling. Full use of nominal and effective interest is allowed for both discrete and continuous compounding, as well as up to four interest regions. Figure 7 displays an Interest spreadsheet that has been automatically expanded by Tutor in response to a problem's input data. Its entries show that the problem data can be given in either nominal or effective terms for commonly used compounding periods, such as continuously, monthly, or daily compounding. Also shown is an example of the data entry guidance that can be triggered by selecting an input cell.

Diagram Period	Rate Region Name	Ends	(Stated) Interest	(Stated) Period	Nom / Eff	Compound Every	Rate to Calculate Eff Rate	Effective Rate for Diagram
Year	A	20	9.5000%	Year	Nom	Cont		
	B	60	10.0000%	Year	Nom	Month		
	C	180	2.5000%	Quarter	Eff	Day		
	D	Inf	0.7500%	Month	Eff	Month		

Rate per Compounding Period

Please put formula in terms of iB.

Example 1: Stated Interest: iB
 Stated Period: Quarter, Nominal
 Compound Every: Month
 Enter in cell: =iB/3

Example 2: As above, but iB is effective
 Enter in cell: =(1+iB)^(1/3)-1

Cell name: ceB

Figure 7. Sheet Interest and the Office Assistant

System limitations include an integral time scale from -10 to 240, although infinite series or multiperiod series are allowed to continue after time 240. Worksheet Amounts provides fifteen columns for storing data. Tutor always places series and gradients in the same column to facilitate referencing them, and it will reuse storage locations that contain 0's. Sheet Main can accommodate the entry of twenty equations, and as many Undo's as needed are allowed. In cases where command buttons have multiple options, an Office Assistant is used to display them.

Summary and Conclusions

The intent of the Equivalence Tutor is to provide distance students with assistance similar to that traditionally offered during a professor's office hours. It also can assist students receiving traditional lectures who find it difficult to use office hours. Smiley faces, congratulatory displays, and programmed prompts are not the same as the warmth of a caring professor, but Tutor is not hurried and it is very patient.

The breadth of Tutor's capabilities should not pose an obstacle to students who only need to learn how to use one factor at a time, starting with the easy ones. An initial demonstration during lecture is complemented by Tutor's prompt, error checking, and help systems: Most students to learn how to use a computer system by experimenting and using the help facility, not by reading a manual. The consistent interface allows a natural expansion from one factor to the next, and the separation of factor use from issues of compounding frequency allows students not to worry about that dimension of the course until it is covered in class.

A prototype has established the feasibility of all of the design features, and coding of the full system is underway with a target date corresponding to the ASEE Conference. Professors wanting a copy of Tutor and any updates of this paper can contact the author at ristroph@louisiana.edu.

Bibliography

1. Terpenny, Janis P., Kimberly Sward, and William G. Sullivan, "Virtual Classroom for Teaching the Economics of Engineering Design," *Proceedings of the 2001 American Society for Engineering Education Annual Conference*, http://www.asee.org/conferences/search/01121_2001.pdf.
2. Terpenny, Janis P., William G. Sullivan, Harpreet Singh, and Kimberly Sward, "Utilizing the Internet to Improve Student Learning in a First Course in Engineering Economy with Real-World Unsolved Problems in Collaboration with Industry," *Proceedings of the 2002 American Society for Engineering Education Annual Conference*, http://www.asee.org/conferences/caps/document2/2002-1160_Paper.pdf.
3. URL for Virtual Classroom for Economics of Eng. Design: http://mielsvr2.ecs.umass.edu/virtual_econ/.

Biography

Dr. John H. Ristroph is a registered professional engineer who serves as Professor of Engineering and Technology Management in the College of Engineering at the University of Louisiana at Lafayette. He is also Professor of Economics. His B.S. and M.S. are from LSU, and his Ph.D. is from VPI&SU, all in industrial engineering. He has taught engineering economics for over thirty years, and enjoys bringing his research into the classroom. His practical experience includes serving as Director of Policy and Planning for the Louisiana Department of Natural Resources where he performed numerous large-scale economic analyses.