Reinforcing Learning in the Data Communications Course Using a Teleprocessing Line Speed Decision Support System

Donald A. Carpenter
Department of Management and Marketing
University of Nebraska at Kearney
Kearney, NE 68849 USA
doncarpenter@unk.edu

ABSTRACT

Two challenges exist in the typical data communications course. First, most traditional students have had very little technical networking experience. Consequently, they lack a practical framework to synthesize all the detail contained in a basic data communications course. Second, the line speed formula taught in many courses is too simplistic to be practical. The formula does not include all the factors it should include such as the impact of noise or overhead, message processing or queuing time, or need to deal with multiple message types and lengths and with peak periods. Consequently, students receive the wrong impression. A great opportunity exists to improve student learning in the data communications class by using a much more complete formula and a method to incorporate the formula into a teleprocessing line speed decision support system. That can also provide the basis for several student projects to reinforce their learning of the many interrelated data communications topics. This paper presents the design of such a teleprocessing line speed decision support system as well as student outcomes in data communications courses.

Keywords: teleprocessing, line speed, decision support system, data communications

1. INTRODUCTION

A telecommunications line is a critical factor in any teleprocessing system. The transmission capacity, or line speed, of that facility should be well matched to the volume of messages that travel in the teleprocessing system. If the line speed is too slow, communications delays and user frustrations result. If the line speed is too fast, the organization will overpay for under-utilized resources. Such problems are not quickly corrected due to cost of changing to a more appropriate teleprocessing line speed, as well as to contractual obligations that might exist for an existing line.

A simple line speed formula (Equation 1) can be used to provide a close estimate in many cases (Green 1996).

$$X = L / T \qquad (1)$$

where X is the line speed, in bits per second (bps),
L is length of message to be transmitted, in bits,
and T is time desired for transmission, in seconds.

Far too often, firms select teleprocessing line speeds without paying sufficient attention to the many other variables that should influence the proper choice. Often, a teleprocessing system operates well even though the owning firm chooses a less than ideal line speed. That is entirely possible, due to line speeds that are available from telecommunications suppliers. Intervals between the available line speeds are large enough that organizations select lines that have sufficient excess capacity. Another saving grace is that the line speed used by some popular network protocol is so incredibly fast that requirements of nearly any user organization could be covered by excess capacity.

Conversely, risks are too great to leave such important decisions to chance. Excess capacity translates to excess cost. Too little capacity means to inefficient operation. Due to risks involved, teleprocessing managers ought to explore thoroughly the line speed issue to insure that the chosen facility is an appropriate match to organizational needs.

Equation 1 does not include all critical variables that should be considered in order to insure acquisition of an appropriate line speed for present and future system needs. Moreover, students should be knowledgeable and experienced with a more complete formula for calculating line speed. If students are only familiar with Equation 1, the problem self-perpetuates in businesses.

2. OFTEN IGNORED VARIABLES IN LINE SPEED CALCULATION

The size of the message to be transmitted and desired lapsed time for transmission are two of the most critical factors in determining the required line speed (Stamper 1999), as reflected by Equation 1. However, that does not consider all the many variables that can confound the line speed decision (Carpenter 1992; Green 1986) which stem from four sets of factors as explained in the following subsections.

2.1 Impact of Interactive versus Batch Systems

Equation 1 does not address the degree of interactivity of the overall teleprocessing system (Green 1986) or the ensuing impact on message volume. In a predominately batch operational environment, throughput is the most important measure of desired transmission time (Stamper 1999). For example, as users create batch files in a heads-down data entry or program development environment on a centralized file server or mainframe computer, the message flow might be predominately unidirectional from the terminals to the processor. Each logical message (i.e. messages perceived by users) usually corresponds to one physical message flowing across the teleprocessing channel.

By contrast, in an interactive or conversational systems setting, such as an inquiry-response system or in an electronic mail system, the more appropriate measure of desirable transmission duration is response time (Daigle 1992). In such environments, each logical message sent by a user results in a physical message that represents the user's inquiry plus at least a second physical message that represents the response generated by the responding node or user. The size of the logical response message is determined by the nature of the inquiry and amount of information that satisfies the initial logical message. Therefore, in an interactive setting, determining the size of the message to use in the line speed calculation is a more complex task.

In reality, most teleprocessing systems include some mix of both batch and interactive messages. Furthermore, there most likely exist several variations of each type of message. Before any line speed calculation formula is applied, one should first determine the impact of the variety of message types and sizes. It is not a straight forward process as considerable analysis is required for the typical teleprocessing system. The teleprocessing system analyst must consider that the system most likely utilizes more than one message length or packet length (Greene 1986). Of course, under some strict data comm protocols, such as X.25, there is a singular standardized packet length (Spragins 1991). One prescribed technique to deal with a variety of message lengths is to determine an average message length (Chou 1974). That is done by summing the products of the sizes of each of the message type times the quantity of that message types and then dividing by the total number of messages.

2.2 Peak Periods Impact on Average Message Length

An additional is whether the mix of messages is constant at all times of operation of the teleprocessing system. On most systems, the message traffic patterns vary within the course of the day, week, month, quarter or year. As a result, the analyst must pay close attention to the peak periods of message traffic (Held 1983). It is critical that a teleprocessing system be designed so that it provides an acceptable throughput or response time for the ultimate peak period(s). If the peak period's demands are met, the system will be adequate for all non-peak periods as well.

Examination of peak periods might provide insight and opportunities to manage system usage patterns in order to alter peak periods. For example, analysis of potential or existing message traffic patterns might indicate that an application, such as entering general ledger adjusting entries, typically occurs in mid-morning. That might coincide with the peak period for interactively entering customer telephone orders in real time. In order to minimize the line speed required to handle the combined impact of the two message types, management might choose to defer the general ledger entries and reschedule for a less hectic time of the day.

Obviously, the system analyst must collect and examine a considerable amount of data in order to determine average message length. The best sources of that data are system users and, if available, a computer resource accounting systems (Held 1983). The task can be complex and tedious. Yet, in order to accurately and thoroughly determine appropriate telecommunications line speed, data about message lengths, quantities, distributions, destinations, peak periods and priorities must be collected and analyzed (Chou 1974).

2.3 Impact of Overhead

Another factor not considered by Equation 1 is overhead (Carpenter 1992, Green 1986). American National Standards Institute (ANSI) recommends several formulae for determining transfer rate of information bits (TRIB). The formulae indicate the set of information bits, i.e. those that represent the user's logical message, are a subset of any physical message (Carpenter 1992, Green 1986). That is to say that, in addition to the bits that represent the logical message as perceived by users of the system, there always also exists a set of bits associated with message overhead.

Overhead bits may be required to propagate the message (i.e., cause it to flow) or can result from special routing or loading factors (Spragins 1991). Most protocol, for instance, include message-polling bits, message-framing bits, message identifying bits, etc. Under some protocol, overhead bits are expressed in terms of characters that must be converted to bits. The conversion of characters to bits is dependent on the number of bits per character in the coding scheme employed. Some coding schemes, such as ASCII, include additional overhead bits for parity detection.

A major classification of overhead relates to error detection and recovery. Every telecommunications facility is subject to interference, a.k.a. noise. There are many techniques for reducing noise but no facility is devoid of noise. Noise can result in changed bits, or errors in the transmitted data. Often, techniques used to detect and correct errors require retransmission of part or all of a message. Such error related retransmissions are also classified as overhead and should be accounted for in any calculation of a required teleprocessing line speed (Carpenter 1992).

2.4 Impact of Message Congestion

A last factor not addressed by Equation 1 is the impact of message congestion on a teleprocessing line. In multi-user systems, the potential exists that multiple users will concurrently attempt to transmit messages. In those instances, there will be contention for use of the communications line, which will result in messages waiting in buffers to access the line. Often, the greatest portion of total response time is due to queuing of messages (McGregor 1974). Queuing formula could feasibly be used to model such instances (Carpenter 1992; Martin 1972).

Application of queuing theory in the design of teleprocessing systems is recommended for a wide variety of situations (Martin 1972). Some of the very first applications for queuing theory were for telecomm facilities. It has been suggested that even the most complex telecommunications networks can be modeled as a series of independent queues (McGregor 1974; Spragins 1991) with queuing theory applied serially to each queue. While queuing formulae do not necessarily yield exact results, they are reasonably accurate for determining line speed (Green 1986; Martin 1972).

Of the dozens of queuing formulae, the most appropriate for single-server queues is the Polloczek-Khintchine (P-K) equation that is applied to M/G/1 queues (i.e. single server queues with exponential service times, a general service discipline, infinite system capacity, and FIFO service order) (Martin 1972). The P-K equation is valid for any message service time distribution, including complex computer-based polling schemes (McGregor 1974; Stamper 1999). The P-K equation assumes exponential message interarrival times (Martin 1972), a condition that typically exists when there are a large number of independent users accessing a teleprocessing system (Tannenbaum 1981). In situations where the P-K formula should not be applied, some other queuing equation could be substituted (Daigle 1992). For instance, it would not be appropriate to use the P-K formula with rigid priority schemes, frequent interrupts, multiple parallel servers, or deterministic service times (Martin 1972).

As queuing theory is applied, one should always pay attention to the rate of utilization of the teleprocessing system. Utilization can be calculated by multiplying the average number of message arrivals per second by the average message service time in seconds (McGregor 1974). Utilization cannot reach 100% or queues will grow indefinitely (Martin 1972). Preferably, utilization level should not exceed 70 or 80%. To lower utilization rate of the complete system, service rate of the system can be increased or the arrival rate of the messages can be decreased. An increase in the teleprocessing line speed could also be an alternative (McGregor 1974).

3. APPROPRIATE LINE SPEED CALCULATION

Equation 2 presents a more appropriate model on which to base the calculation of the line speed for a multi-user teleprocessing system. It includes all those variables that are missing from the simplest line speed formula, Equation 1. A thorough explanation of the derivation of that equation can be found in Carpenter 1992. Its use for decision-making is found in Carpenter 1993.

$$X = \frac{(1 + Y)(1 + N)\max_{j=1}^{k} \dfrac{\sum_{i=1}^{n} (((L + H_c)C + H_b)_{ij}\, m_{ij})}{\sum_{i=1}^{n} m_{ij}}}{\mathbf{R - S -} \dfrac{AS^2}{Z\,(1 - AS)}} \qquad (2)$$

where X is needed line speed, in bits per second (bps),
- L is length of each anticipated logical message type, in characters,
- m is quantity of a logical message type per period,
- n is total number of messages in each time period,
- k is the total number of time periods examined (the MAX function will find the peak period),
- i and j are indexes, varying from 1 to n and 1 to k,
- H_c is overhead for a message type in characters,
- H_b is any overhead for a message type in bits,
- C is the conversion factor of bits per character,
- Y is a constant from 0 to 1 that reflects a proportion of messages that require replies,
- N is a non-negative factor for the average percent of retransmissions required due to noise on line,
- R is users' required response or throughput time,
- S is the average message service time in seconds,
- A is average message arrivals per second, and Z is a variable ranging from 1 to 2 representing observed degree of variability in the service time (Value of 1 would be best-case scenario; 2 would be the worst case. This might not be easily observed, the equation should be run using both extremes and the results compared.)

One major precaution must be taken prior to applying

Equation 2. The utilization of the network must be between 0 and 1 for the system to function; preferably between .7 and .8. Utilization is calculated as shown in Equation 3.

$$U = AS \qquad (3)$$

where U is the utilization percentage,
S is average message service time in seconds,
A is average message arrivals per second.

4. STUDENT ASSIGNMENTS BASED ON A TELEPROCESSING LINE SPEED DECISION SUPPORT SYSTEM

Each variable in Equation 2 has been discussed in the literature for over three decades, so solutions methods have been forthcoming. Computer simulation is one way of modeling the interaction of the variables (Held 1983). However, cost and complexity of model building and the lack of expertise by many firms result in such methods being more rarely applied than they could be (Martin 1972). Similarly, commercial network design and simulation packages exist for the application (Chou 1974). Unfortunately, those network design packages are typically priced out of the range of most small businesses and educational institutions. Furthermore, most of those software packages execute on hardware platforms that are larger than the systems available to most small businesses and schools (Chou 1974).

Obviously, Equations 2 and 3 can be solved manually. However, time required to solve the equation increases proportionately with complexity of the system being designed. Consequently, to perform and double-check the calculations manually for a complex network would require a large amount of time. "What-if" analysis is tedious if done manually. Since calculations and sensitivity analysis can be done more efficiently using a computer, idea arose for teleprocessing line speed decision support system (TLSDSS) (Carpenter 1993).

A teleprocessing line speed decision support system obviously would include both Equations 2 and 3, plus a flexible user interface and what-if analysis capabilities necessary to make sound decisions (Carpenter 1993). Such a TLSDSS could also be used as a meaningful programming assignment in upper level courses in data communications and distributed processing, as well as a pedagogical tool in computer literacy and other courses. The following sections present design specification for that programming assignment and more detail about the TLSDSS as a pedagogical tool.

5. DESIGN SPECIFICATIONS FOR THE TLSDSS

5.1 Overview and User Interface

A menu approach to the user interface (whether a simple list of options or point-and-click icons) would portray a modular design to the TLSDSS. One possible menu structure is given below. The first five menu choices are elaborated in subsequent sections.

There might be six main menu choices. A first menu choice would allow input of the data used to calculate average message length. A second menu choice allows entry of data that not related directly to message sizes and volumes. A third menu choice calculates and outputs teleprocessing line speed solution and related information. A fourth menu option enters the portion of TLSDSS that performs what-if and sensitivity analysis at the operator's discretion. The fifth choice displays line graphs to illustrate relationship among variables. From the main menu, the user could also enter the help facility via a sixth menu choice. Help that is available from the main menu is a general description of the use and purposes of the TLSDSS. That help is couple brief pages presented in paragraph format. Help for other parts of the program are specific to the tasks at hand.

5.2 Average Message Length Calculations

The bulk of data input into the TLSDSS relates to the calculation of the average message length. That portion of the input data is typically collected by interviewing users or by observing the existing system in operation. Therefore, it is logical that the process of inputting that portion of the data is segregated from the input of the remainder of the data (Table 1). That segregation is enforced by a main menu selection in the TLSDSS for these data, separate from the data explained below

5.3 Modularizing Line Speed Calculation Formula

Equation 1 provides the primary engine for TLSDSS. However, the equation is not incorporated intact in TLSDSS. Rather, Equation 1 is subdivided into logical parts. In that manner, changes can be made to some if the variables without requiring recalculation of the entire equation. Table 2 illustrates logical subdivisions of the formulae as they are embedded in the TLSDSS. Subdividing Equation 1 serves two other important roles in addition to facilitating what-if analysis. First, it gives a useful learning tool for students to more thoroughly understand the formula. Second, if the TLSDSS were to include an explanation facility, the subdivisions of the equation would be logical boundaries around which such a facility could be built.

5.4 Inputting Basic Data

By selecting the first menu option, the user can enter the data required to determine the average physical message length in the peak period. TLSDSS allows entry of data for a large number of periods, for a large number of users and logical message types for each user. Logical messages are numbered so that volumes can be tracked and totaled by message by users, providing a set of data for analysis at the decision-maker's discretion. A sample of the input data required to determine average message length was provided in Table 1. The format of the table is similar to the layout of the input form in the TLSDSS. One difference is that the data in the table is only for one period. The TLSDSS actually goes through an iteration for each of a large number of

Table 1: Sample Input Data to Determine Average Message Length in Peak Period

User	Message Type	Message Quantity	Logical Message Length	Control Characters	Bits per Char	Contro l Bits	Physical Message Length	Impact (quantity X length)
A	1	600	900	43	8	16	7560	4,536,000
	2	200	480	43	8	16	4200	840,000
	3	1300	300	43	8	16	2760	3,588,000
Subtotal		2100						8,964,000
B	1	1200	900	43	8	16	7560	9,072,000
	2	300	480	43	8	16	4200	1.260,000
Subtotal		1500						10,332,000
C	2	700	480	43	8	16	4200	2,940,000
	3	900	300	43	8	16	2760	2,484,000
Subtotal		1600						5,424,000
D	2	600	480	43	8	16	4200	2,520,000
	3	1000	300	43	8	16	2760	2,760,000
Subtotal		1600						5,280,000
E	3	400	300	43	8	16	2760	1,104,000
Subtotal		400						1,104,000
Total		7200						31,104,000

Table 2: Logical Subdivisions of the Line Speed Equation

Factor	Portion of the Equation
Physical message length	$((L + H_c) * C + H_b)$
Impact of each message	(physical message length)$_{ij}$ * m_{ij}
Cumulative impact of messages in a period (CIMP)	$\sum_{i=1}^{n}$ (impact of each message)
Cumulative number of messages in a period (CNMP)	$\sum_{i=1}^{N} m_{ij}$
Average physical message length	CIMP / CNMP
Average physical message length in the peak period (APMLPP)	$\max_{j=1}^{K}$ (average physical message length)
Impact of messages requiring answers (IMRA)	(1 + Y)
Impact of noise (IN)	(1 + N)
Maximum number of bits per average user request (MNBPUR)	IMRA * IN * APMLPP
Portion of total time due to message queuing (Q)	$[A * S^2] / [Z * (1 - A * S)]$
Net time a message spends on line (NTMSOL)	R - S – Q
Required line speed	MNBPUR / NTMSOL

Table 3 Sample of Remainder of Input Data and Calculated Ouputs

Factor	Data
Calculated Peak Period Average Physical Message Length (in bits) = (((average logical message length + control characters) X conversion factor) + control bits) / message quantity = cumulative impact / cumulative quantity = 31,104,000 / 7200	4320
Average Number of Messages Arriving per second during peak hour = total # of messages in peak hour / total seconds per hour = 7200 / 3600	2.0
Line Noise Factor (number > 0, representing percent of retransmissions)	1.0
Percent of Messages Requiring a Response (a factor between 0 and 1)	1.0
Enter Average Service Time per message (in seconds)	.025
Observed Degree of Variability in Service Time (between 1 and 2)	1.0
Desired Response Time (in seconds)	5.0
THE CALCULATED LINE SPEED, in bits per second	3,474.2
Calculated System Utilization	5.0%

periods. In that manner, the data can be analyzed by the TLSDSS for each period separately in order to determine the peak period. Designing the network for the peak period will allow sufficient slack for the network to be able to handle all periods.

Table 1 illustrates input for five users (A - E) and up to four message types per user. Each user has a different pattern of message usage and a different volume of each type of message. For each user-message combination, data is entered to indicate the overhead characters, the number of bits per character in the coding scheme being used, and the number of additional bits of overhead. The TLSDSS allows for each user-message combination to have a different set of values for that data. If the decision maker does not specify the values, the TLSDSS repeats the last set of values entered. In the interest of simplicity, Table 1 only illustrates one set of those three values. The TLSDSS provides subtotals by user and message type plus grand totals.

5.5 Inputting Remainder of Data

The second option from the main menu is to enter the input data that does not affect the calculation of average message length. There are seven entries that can be made. Only five of the entries must be made, as the TLSDSS will have calculated the other two. A sample set of those seven variables in presented in Table 3.

Using previously entered data that affects the calculation of average message length, the TLSDSS will perform that calculation. Therefore, the decision maker need not reenter that data . Likewise, a TLSDSS will calculate the number of message arrivals per second, eliminating need to enter that data.

There are two occasions when the decision maker might choose to enter the average physical message length and/or the number of arrivals per second rather than use the values calculated by the TLSDSS. One of those occasions is when the decision maker has not already entered the data that the TLSDSS uses to perform those calculations. For instance, a decision maker might have derived or might be estimating those items without collecting all the raw data.

The other occasion is when a decision maker wants to perform a sensitivity or what-if analysis. Calculated data can be noted along with calculated line speed, then overridden by entering other values. A decision maker can get a feel for impact of changes in values on calculated line speed.

The other five values to be input are (1) a line noise factor which is a number greater than zero, representing percent of retransmissions due to noise on the telecom line, (2) percent of messages requiring a response, which is a factor between zero and one, (3) the average service time per message by a central processing unit, expressed in seconds, (4) observed degree of variability in service time, a factor between one and two, and (5) desired response or turnaround time, given in seconds.

Typically, the first four of those factors are determined through conversations with the technical staff or with computer system vendors. Alternatively, a reasonable estimate could be substituted for any of the four factors. Management specifies desired response, often given as a rigid teleprocessing system design constraint.

5.6 Calculating the Line Speed

After entering all input data, the next logical step is to select the option from the main menu that calculates and displays the required line speed for the teleprocessing line. That option displays input data as well as calculated line speed. Table 3 illustrates that.

In addition to calculating and displaying the required line speed, the TLSDSS also calculates and displays the system utilization rate. The ideal utilization range is between seventy and eighty percent. Therefore, the TLSDSS displays a warning message if the calculated utilization rate falls outside that range. If the calculated utilization rate is greater than or equal to one hundred percent, the TLSDSS displays a different message that the calculated line speed is invalid, as utilization cannot reach or exceed one hundred percent.

After the TLSDSS displays the input and output data, an option is available to a decision maker. The choice can be made to add the currently displayed data to a table for storage. In that manner, the decision maker can collect data from several iterations of input and calculation for analysis at a later time. Choosing the analysis option from the main menu can access that table of data. Those analyses are explained below.

5.7 What-If and Graphical Analysis

The fourth and fifth main menu selections provide a variety of ways to perform analyses on the data. One set of options is to view the data in several tabular presentation modes. The other set of options is to view the data in several graphical presentation modes. The TLSDSS reminds the operator as to how to change the message size and volume data. Basically that is a matter of returning to the main menu and selecting the option that allows for that data to be input again. The last set of data entered in the current session will still be available for perusal and change as appropriate.

The TLSDSS also informs the operator that sensitivity analysis can be performed on the other input data as well. The system will recalculate as many variations on the input data as the operator cares to provide. By selecting the appropriate option after the calculation of the line speed, the operator can direct all that data to be stored in a table. There is an option available on the screen that allows the operator to view that table.

In addition to those analyses, the TLSDSS also will solve the equation for variables other than line speed.

To choose that option, the operator must provide the teleprocessing line speed for the TLSDSS to use in deriving the other solutions. The TLSDSS will use all the most recently input and output variables to solve for the specified variable. The TLSDSS can produce several graphs for visual analysis as shown by Table 4.

Table 4: Graphical Analyses Possible by TLSDSS

Graph Type	Dependent Variable	Independent Variable(s)
Line	Utilization	Line Speed
Line	Line Speed	Mean Message Arrival Rate
Line	Line Speed	Average Message Length
Line	Line Speed	Mean Message Service Time
Line	Line Speed	Service Time Variance
Line	Line Speed	Degree of Noise on the Line
Line	Line Speed	Desired Response Time
Line	Line Speed	Proportion of Messages Requiring Answers

6. LIMITATIONS

Six idiosyncrasies might exist in some teleprocessing systems would limit applicability of the TLSDSS. That is due to the fact that the Polloczek-Khintchine queuing equation is not applicable for all teleprocessing systems. Other queuing formulae might be more appropriate for systems that exhibit those characteristics (Daigle 1992, Martin 1972). The TLSDSS would need to be altered to incorporate those formulae in order for the TLSDSS to be used in conjunction with such systems.

First is existence of a priority scheme that is elaborate in nature or that is rigidly enforced. Priority schemes tend to enforce other than first-come first-served service disciplines. For example, a least-recently-served-first or a shortest-processing-time-first service discipline would invalidate use of the P-K queuing equation.

Second is the occurrence of frequent interrupts of its service. Interrupts can be caused by faulty equipment or intentionally, for instance, by CPU-activated automatic dialing systems.

Third is possible existence of multiple parallel servers as in a parallel processing setting. If any one of several CPUs can provide service for each message, then P-K formula might not be valid. In some instances, multiple parallel servers can be modeled by using an average total service time required for a message to be completely handled by the entire set of processors. If that can be done, then the Polloczek-Khintchine queuing equation might still be applicable.

Fourth is case of dependent service times. For instance, if service time is reserved for user messages as with assigning prearranged times on a dial-up system, then the Polloczek-Khintchine equation should not be used.

The fifth idiosyncrasy occurs in many complex systems with multiple serial servers. Very often, the cumulative impact of multiple serial servers can be modeled as if there was only one server. Other times the model used in the TLSDSS can be applied successively to each of the multiple serial servers.

Sixth is the variable treatment messages might receive on an integrated services digital network (ISDN). By combining classical data communications messages with voice, video, and facsimile transmission (FAX), the nature of teleprocessing system changes considerably and might invalidate the use of the P-K formula. If each message types can be quantified in terms of bits per physical message, then the TLSDSS model as presented above might readily apply to an ISDN.

7. CONCLUSIONS: TLSDSS CLASSROOM USE

The teleprocessing line speed decision support system, in the format described herein has been successfully applied within several realistic teleprocessing systems development projects. In some instances, the line speed calculated by the TLSDSS has influenced the decision makers to change their initial preliminary decision and opt for either a higher or lower speed teleprocessing line speed. In other instances, use of TLSDSS has served to confirm the decision maker's preliminary line speed decision. The value of TLSDSS has been considerable in those live systems projects.

The TLSDSS has proven to be an extremely valuable pedagogical tool for university students. Since 1981, the author has taught a course in data communications and distributed processing. As is customary in such courses, there is heavy coverage of communications terminology and techniques. It was the author's observation that the students typically lacked an appreciation for how the myriad of teleprocessing system design choices were interrelated and how they influenced each other in actual practice. Furthermore, typical textbook examples tend to encourage the use of the very simplest equation to calculate required speed for communications lines and attached components.

That scenario provided the stimulation to develop a more realistic equation and to encourage student use of that equation in case studies. As the equation has evolved to incorporate more variables, so have the requirements for students' use of that equation. A DSS paradigm was seen as an obvious tool to use in the data communications and distributed processing course.

In senior/graduate course for computer information systems, business and telecommunication majors, the assignment is approached in the different manner. First, the equation and its usage are discussed in class. Then, students are encouraged to use macro languages of spreadsheet or database package to create and implement their own version of the TLSDSS. In a similar senior/graduate level data communications course for computer science majors, students are asked to design and implement the TLSDSS algorithm using an appropriate high-level computer language, e.g. C++, Pascal, Java, etc. Creating the DSS for themselves adds an outstanding learning element to the assignment.

The author has used the TLSDSS in other educational settings. In an introductory computer literacy course that enrolls students from a wide variety of academic disciplines, TLSDSS has been supplied to the students as a ready to use package. Similarly, in a graduate level "educational technology" course for on-the-job primary and secondary teachers, the TLSDSS has been given to students, who use it extensively and report on the relationship among the variables.

The copy of the TLSDSS provided to those students was a recently written version by a student in a more advanced class. In some cases, that planted the seed for some beginning students to consider switching to major in computer science and information systems. A copy of such a student-written TLSDSS in MS Access format can be requested from the author via email.

Student learning experience has been invaluable. There has been a measurable increase in the students' levels of understanding of basic teleprocessing concepts and interrelationships of the large number of variables. There has been strong positive feedback from students as to the perceived value to them of this approach to the material. Employers of the students have also responded favorably. In several instances, alumni have reported that their experience with the TLSDSS has made the difference in securing initial employment.

8. REFERENCES

Carpenter, Donald A. [1992], "A Queuing Model for Multi-user Teleprocessing System Line Speed Decisions." Proceedings of the 1992 ACM Computer Science Conference, March, pp. 407-414.

Carpenter, Donald A. [1993], "Improving Quality of Teleprocessing Line Speed Decisions." Proceedings of 24th Annual Meeting of Midwest Decision Sciences Institute, April, pp.

Chou, W. [1974], "Planning and Design of Data Communications Networks." Proceedings of the National Computer Conference.

Daigle, J.N. [1992], Queuing Theory for Telecommunications. Reading, MA: Addison Wesley Publishing Company.

Green, J. [1986], Microcomputer Programs Can Be Adapted for Data Network Design." Data Communications, April, pp. 116-128.

Held, G. [1983], "No More Guesswork for Sizing Network Components." Data Communications, October, pp. 199-211.

Martin, James [1972], Systems Analysis for Data Transmission. Englewood Cliffs, NJ: Prentice-Hall, pp. 413-480.

McGregor, P. [1974], "Effective Use of Data Communications Hardware." Proceedings of National Computer Conference.

Spragins, J.D., Hammond, J.L. and Paulikowski, K. [1991], Telecommunications: Protocol and Design. Reading, MA: Addison-Wesley, pp. 231-242.

Stamper, David A. [1999], Business Data Communications, 5th Ed. Redwood City, CA: Benjamin Cummings, pp. 193-194.

Tanenbaum, Andrew. [1981], Computer Networks. Englewood Cliffs: Prentice-Hall, pp. 56-67.

AUTHOR BIBLIOGRAPHY

Donald A. Carpenter is Professor of Management Information Systems and Director of the Global Sources Information Technology Programs at the University of Nebraska at Kearney. He holds a bachelors degree from Kearney State College, an MBA from the University of Colorado at Colorado Springs, and a Ph.D. from the University of Nebraska-Lincoln. Teaching and research interests are in CIS education, decision support systems Information requirements determination and meaningfulness of IS work. He has published in *Journal of Computer Information Systems, Journal of Computer Science Education, International Journal of Decision Support Systems,* and others.

Information Systems & Computing
Academic Professionals

STATEMENT OF PEER REVIEW INTEGRITY

All papers published in the Journal of Information Systems Education have undergone rigorous peer review. This includes an initial editor screening and double-blind refereeing by three or more expert referees.

ISSN 1055-3096