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

Donald A. Carpenter
Management and Marketing
University of Nebraska at Kearney
doncarpenter@unk.edu

Abstract

Two challenges exist in the typical data communications course. First, most traditional students have had little technical network. Consequently they lack a practical framework to synthesize all the detail contained in a basic data communications course.

Second, the line speed formula (X=L/T) 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 the need to deal with multiple message types and lengths and with peak periods.

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 (TLSDSS). That TLSDSS can also provide the basis for several different student projects to reinforce their learning of the many interrelated data communications topics.

Introduction

A critical factor in any teleprocessing system is the communications line. The transmission capacity or speed of that line must be well matched to the volume of messages that travel throughout 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 an under-utilized resource. Such problems are not usually corrected quickly due to the cost of changing to a more appropriate teleprocessing line speed. Contractual obligations pertaining to the existing line might also increase the difficulty of solving the problem quickly.

Far too often, firms select their teleprocessing line speeds without paying sufficient attention to the plethora of variables that should influence the proper choice. Often, a teleprocessing system operates reasonably well even though the owner firm chose a non-ideal line speed. That is entirely possible, due to the line speeds that are available for acquisition from communications common carriers. The intervals between the available line speeds are large enough that many organizations select lines that have sufficient excess capacity to allow for proper system operation. Another saving grace is that the line speed of some popular network protocol is so incredibly fast that the requirements of nearly any user organization could be covered by the excess capacity.

Conversely, the risks are too great to leave such an important decision to chance. Excess capacity can translate to excess cost. Insufficient capacity will translate to inefficient operations. Due to the 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. Unfortunately, the most common quantitative approach to the problem is to use a greatly over simplified formula.

It is commonly assumed that to calculate the appropriate teleprocessing line speed, one divides the size of the message to be transmitted (in bits) by the desired message transfer time (in seconds). The result is assumed to be the proper line speed, expressed in bits per second. Indeed, message size and transfer time are two of the most important variables that should be considered. Hence, a formula that includes only those two variables can provide a reasonable estimate of the appropriate line speed. That also accounts for part of the success of those teleprocessing systems that have been designed by only using that simplest formula. However, that simplest formula does not include all the critical variables that should be considered in order to insure acquisition of an appropriate line speed.

The next section describes the variables which are typically not considered, but which ought to be included in a properly made line speed decision. The importance of the variables is based on a review of the relevant literature. Subsequent sections describe the creation and application of a teleprocessing line speed decision support system that incorporates all the important variables.

Typically Ignored Variables in Line Speed Calculation

As explained above, the size of the message to be transmitted and the desired lapsed time for transmission are two of the most important factors in determining the required line speed for a teleprocessing system (Stamper). The combined impact of those two factors can indeed provide a reasonable approximation of the necessary line speed. An often-used formula is to divide the message size by the transmission time (Green). However, that formula does not consider all the many variables that can confound the line speed decision (Carpenter 92, Green). To overlook variables other than transmission size and message size can result in an improper teleprocessing system design [Carpenter 92).

First of all, that simple equation does not address the degree of interactivity of the overall teleprocessing system (Green) and the resultant impact on the volume of messages. In a predominately batch operational environment, throughput is the most important measure of desired time for transmission (Stamper). 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. That is to say, each logical message as perceived by users usually corresponds to one physical message flowing across the teleprocessing channel.

By contrast, in an interactive or conversational environment, such as an inquiry-response system or in an electronic mail system, the more appropriate measure of desirable transmission duration is response time (Daigle). 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 the 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 classical 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 nearly a straightforward process: considerable analysis is required for the typical teleprocessing system. The analyst of a teleprocessing system must consider that the system most likely utilizes more than one message length or packet length (Greene). Of course, under some strict data communications protocols, such as X.25, there will be one singular standardized packet length (Spragins). One prescribed technique to deal with a variety of message lengths is to determine an average message length (Chou). That is done by summing the products of the sizes of each of the message types times the quantity of that message type and then dividing by the total number of messages.

An additional consideration that should be made in determining the average message length is whether the mix of messages is constant at all times of operation of the teleprocessing system.

On most teleprocessing 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). 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.

The examination of peak periods might provide insight and opportunities to manage system usage patterns in order to alter the peak periods. For example, the 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 them for a less hectic time of the day.

Obviously the 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). The task can be complex and tedious. Yet in order to accurately and thoroughly determine the appropriate telecommunications line speed, data about message lengths, quantities, distributions or destinations, peak periods and priorities must be collected and analyzed (Chou).

Another set of factors that are typically not considered by the most simplistic line speed formula pertains to the impact of overhead (Carpenter 92, Green). 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 teleprocessing message (Carpenter 92, Green). 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). Most formal protocol, for instance, includes message-polling bits, message-framing bits, message identifying bits, etc. Under some protocol, the overhead bits are expressed in terms of characters that must be converted to bits. The conversion of characters to bits is dependent upon the number of bits per character in the coding scheme employed. Some coding schemes, such as ASCII, include additional overhead bits for parity detection purposes.

A major classification of overhead is that which relates to error detection and recovery. Every telecommunications facility is subject to interference, known as 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 92).

A last set of factors not addressed by the simplest line speed calculation formula pertains to 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 the total response time or throughput time is due to queuing of messages (McGregor). Queuing formula could feasibly be used to model such instances (Carpenter 92, Martin).

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

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 (Martin). That formula is valid for any message service time distribution, including complex computer-based polling schemes (McGregor, Stamper). The P-K equation assumes exponential message interarrival times (Martin), a condition that typically exists when there are a large number of independent users accessing a teleprocessing system (Tannenbaum). In situations where the P-K formula should not be applied, some other queuing equation could be substituted (Daigle). For instance, it would not be appropriate to use the P-K formula in the existence of rigid priority schemes, frequent interrupts, multiple parallel servers, or deterministic service times (Martin).

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). Utilization cannot reach 100% or queues will grow indefinitely (Martin). Preferably, the utilization level should not exceed 70 or 80%. In order to lower the utilization rate of the complete system, the 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).

Each variable presented above has been discussed in the literature for over three decades. Yet most organizations fail to consider most of the variables when they use the simplest line speed calculation formula. Of course, there are those enlightened firms that will take advantage of computer simulation or other scientific management techniques (Held). 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).

Alternatively, there are commercially available network design and simulation software packages for this application (Chou). Many of those packages incorporate the all variables that are excluded from the simplest line speed formula. Unfortunately, those network design packages are typically priced out of the range of most small businesses. Furthermore, most of those software packages execute on hardware platforms that are larger than the systems available to most small organizations (Chou).

On a more positive note, all is not lost for those businesses which lack resources to acquire sophisticated network design packages or which lack expertise to build their own models and to apply computer simulation. An appropriately elaborate queuing oriented line speed calculation formula has been published (Carpenter 92). The model addresses all of the variables discussed above and is presented in an easily applicable manner.

The purpose of this article is to present and discuss a teleprocessing line speed decision support system (TLSDSS) which addresses all the typically missing variables. TLSDSS incorporates the elaborate line speed model as well as a flexible user interface and what-if analysis capabilities necessary to make sound decisions (Carpenter 94). The next section describes the components of the TLSDSS. Subsequent sections discuss application of the TLSDSS and its use as a pedagogical tool for computer literacy courses as well as for advanced courses in data communications and distributed processing.

An Appropriate Model For Line Speed Calculations

As stated previously, most often organizations use a teleprocessing line speed calculation formula that is greatly over simplified. That simplistic equation only considers message length and the duration that the message spends on the telecommunications line. While that simplest equation might provide a reasonable estimate of line speed, it leaves too much to chance. It does not include many of the variables that need to be considered to properly make the critical line speed decision.

To provide a basis for comparison, it is important to briefly examine that overly simplified equation. The formula that is most often used for calculating teleprocessing line speed is: $X = L / T$

where X is the line speed, expressed in bits per second (bps),

L is the length of the message to be transmitted, in bits, and

T is the time desired for transmission, expressed in seconds.

A more appropriate model on which to base the calculation of the line speed for a multi-user teleprocessing system is presented below. It includes all those variables that are missing from the simplest line speed formula shown immediately above. A thorough explanation of the derivation of that equation can be found in Carpenter 92. Its use for decision-making can be found in Carpenter 93.

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

where X is the required line speed, expressed in bits per second (bps),

- L is the length of each anticipated logical message type in characters,
- m is the quantity of each logical message type in each time period,
- n is the total number of messages in each time period,
- k is the total number of time periods examined (MAX will use the peak),
- i and j are indexes, varying from one to n or k, respectively,
- H_c is any overhead associated with the message type in characters,
- H_b is any overhead associated with the 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 indicating the average percent of retransmissions required due to noise on the communications line,
- R is the response time or throughput time required by users,
- S is the average message service time in seconds,
- A is the average number of message arrivals per second, and
- Z is a variable ranging in value from 1 to 2 that represents the observed degree of variability in the service time -- a value of 1 would be the best-case scenario; 2 would be the worst case.

There is one major precaution that must be taken prior to applying the equation. The average message service time (S) multiplied by average number of arrivals (A) represents the utilization level of the network, expressed as a factor between zero and one. The utilization rate must be less than one, and should preferably be in the .7 to .8 range. Otherwise, the teleprocessing system being modeled will not function well, if at all.

The formula includes the basic Polloczek-Khintchine queuing equation. It is the most appropriate queuing equation for most teleprocessing systems. However, there are a few systems that cannot be modeled by that equation and another queuing equation should be used. Explanations of those other queuing equations can be found in several references (Daigle, Martin).

Features Of A Teleprocessing Line Speed Decision Support System

Obviously, the teleprocessing line speed equation presented above can be solved manually. However, the time required to solve the equation increases proportionately with the 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. Calculations and sensitivity analysis can be done more efficiently on a computer. Hence the idea for a teleprocessing line speed decision support system (TLSDSS).

The line speed equation provides the primary engine for the TLSDSS. However, the equation is not incorporated in its entirety in the TLSDSS. Rather, the equation 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 1 illustrates the logical subdivisions of the formulae as they are embedded in the TLSDSS.

Table 1: 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	$\sum_{i=1}^{n}$ (impact of each message)
Cumulative number of messages in a period	$\sum_{i=1}^{n} m_{ij}$
Average physical message length	cumulative impact of messages in period/ cumulative number of messages in period
Average physical message length in the peak period (APMLPP)	$\mathrm{MAX}_{j=1}^{k}$ (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	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	R - S – Q
Required line speed	maximum # of bits per avg user request / net time message spends on line

Subdividing the comprehensive line speed equation serves two other useful purposes in addition to facilitating what-if analysis. First, it provides a useful learning tool for one to more thoroughly understand the formula. Second, if the TLSDSS were to include an explanation facility in the future, the subdivisions of the equation would be logical boundaries around which such a facility could be built.

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 2). That segregation is enforced by selection from a main menu of the TLSDSS.

Table 2: Sample Input Data to Determine Average Message Length in Peak Period

User	Mes-sage Type	Message Quantity	Logical Message Length	Control Char-acters	Bits per Ch.	Con-trol Bits	Physical Message Length	Impact (qty 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
Sub	Total	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
Sub	Total	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
Sub	Total	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
Sub	Total	1600						5,280,000
E	3	400	300	43	8	16	2760	1,104,000
Sub	Total	400						1,104,000
Grnd	Total	7200						31,104,000

There might be five other main menu choices. A second menu allows for entry of the data that does not relate directly to the message sizes and volumes. A third menu choice calculates and outputs teleprocessing line speed solution and related information. The fourth menu option enters the portion of the TLSDSS that performs sensitivity analysis at the operator's discretion. The sixth choice from the menu displays line graphs to illustrate the relationship among key variables.

From the main menu, the user can also enter the help facility. The help that is available from the main menu is a general description of the use and purposes of the TLSDSS. That help is a couple brief pages presented in paragraph format. The help from the other parts of the program are specific to the tasks at hand in those parts.

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 is provided in Table 2. 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 a similar iteration for each of a large number of 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 2 illustrates input for five users (A - E) and up to four message types for each user. Each user has a different pattern of message usage. Likewise, each user has a different volume of each type of message. For each user-message combination, data is entered to indicate the number of 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 2 only illustrates one set of those three values. The TLSDSS provides subtotals by user and message type plus grand totals.

The second option from the main menu is to enter the remainder of 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.

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	4320

= cumulative impact / cumulative quantity = 31,104,000 / 7200	
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%**

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

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, the decision maker might have collected or might be estimating those items without collecting all the raw data. The other occasion is when the decision maker wants to perform a sensitivity or what-if analysis. The calculated data can be noted along with the calculated line speed, and then overridden by entering other values. Thus, a decision maker can get a feel for the impact of changes in those 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 the percent of retransmissions due to noise on the teleprocessing line, (2) the percent of messages requiring a response, which is a factor between zero and one, (3) the average service time per message by the central processing unit, expressed in seconds, (4) the observed degree of variability in service time, a factor between one and two, and (5) the desired response or turnaround time, expressed in seconds.

Typically, the first four of those factors are determined through conversations with the technical staff or with computer system vendors. Alternatively, any of those four factors could be substituted by a reasonable estimate. Desired response time is usually supplied by management, often mandated as a rigid teleprocessing system design constraint.

After entering all the 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 the input data as well as the 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 range for the utilization 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 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. That and other options for analysis are explained below.

The fourth and fifth selections from the main menu provide the decision maker with a variety of ways to perform analyses of 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 the operator to peruse 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 from as many variations on the input data as the operator cares to provide. By selecting the appropriate option following 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: A Few Graphical Analysis Options That Could Be Created by the TLSDSS

Graph Type	Factor One	Factor Two, etc.
Line	Line Speed	Mean Message Arrival Rate
Line	Utilization Rate	Line Speed
Line	Line Speed	Average Message Length
Line	Line Speed	Mean Message Service Time
Line	Line Speed	Degree of Service Variability
Line	Line Speed	Proportion of Messages Requiring Answers

Line	Line Speed	Degree of Noise on the Line
Line	Line Speed	Desired Response Time
Pie	Response Time	Line Time, Service Time, Queuing Time

Limitations

Six idiosyncrasies that are present in some teleprocessing systems would limit the applicability of the teleprocessing line speed decision support system as presented above. 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, Martin). The TLSDSS would need to be altered to incorporate those formulae in order for the TLSDSS to be used in conjunction with such systems.

One idiosyncrasy that might be present is the 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 the use of the Polloczek-Khintchine queuing equation.

Another idiosyncrasy that would invalidate the use of the model as presented above 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.

A third idiosyncrasy is the existence of multiple parallel servers as might exist in a parallel processing environment. If any one of several CPUs can provide service for messages, then the P-K formula might not be valid. In some instances, the 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 P-K formula might still be applicable.

A fourth idiosyncrasy pertains to dependent service times. For example, 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.

A fifth idiosyncrasy occurs in many complex telecommunications 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 incorporated in the TLSDSS can be applied successively to each of the multiple serial servers.

The last idiosyncrasy is the variable treatment that 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 the teleprocessing system changes considerably and might invalidate the use of the P-K formula. If each of the message types can be quantified using the common denominator of bits per physical message, then the TLSDSS model as presented above might readily apply to an ISDN.

Application Of The TLSDSS

The teleprocessing line speed decision support system, in its present format described herein as well as in earlier versions, has been successfully applied in 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 the TLSDSS has served to confirm the decision maker's preliminary line speed decision.

The value of the TLSDSS has been considerable in those live systems projects. In addition, the TLSDSS has proven to be an extremely valuable pedagogical tool for students in university courses. Since 1981, the author has annually taught a course in data communications and distributed processing. As is customary for 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 the 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.

As the popularity of decision support systems has increased, the DSS paradigm was seen as an obvious tool to use in the data communications and distributed processing course. Therefore, in the past few years, students in the author's courses have been required to manipulate the telecommunications line speed equation in a DSS format.

The author has used the technique in several different educational settings. In an introductory level computer literacy course that is targeted toward students from a variety of academic disciplines, the 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 secondary teachers, the TLSDSS has been supplied to students and they are expected to use it extensively and report on the relationship among the variables.

In a senior/graduate level course for information systems, business administration and telecommunications majors, the assignment is approached in the different manner. Those students are encouraged to use the macro language of a spreadsheet or database package to implement their own version of the TLSDSS. Creating the DSS for themselves adds an additional learning element to the assignment. In a similar senior/graduate level data communications course for computer science majors, the equation and its usage are discussed in class. Students are then expected to design and implement the algorithm using an appropriate high-level computer language, e.g. Pascal or C.

The student learning experience has been invaluable. There has been a measurable increase in the students' levels of understanding of the basic teleprocessing concepts and the interrelationships of the large number of variables. There has also 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.

Summary

Most often, enterprises use an overly simplified equation to calculate required speed for the communication lines in a teleprocessing system. The inadequate equation is length of a message to be transmitted divided by the required time for that transmission. Many important factors are not included in the simple equation. Missing factors include impact of multiple message lengths and volumes, peak periods, retransmission, physical message overhead, message service time and variance, and message queuing time.

This paper has presented a formula that incorporates the missing factors that can be built into a teleprocessing line speed decision support system. The TLSDSS accommodates the input variables for all but the largest teleprocessing systems and provides several ways to accomplish what-if and sensitivity analysis in order to support efforts of the decision maker as well as the learning experience of students. Also explained are idiosyncrasies that might exist in a teleprocessing system that would cause changes in the formula.

TLSDSS has been successfully applied in the design of several teleprocessing systems. It has also been used as a pedagogical tool for students in computer literacy environments. Furthermore, the design of the TLSDSS has been successfully used in advanced data communications courses as the basis for effective student assignments.

References

Carpenter, D.A. (1992.) A queuing model for multi-user teleprocessing system line speed decisions, *Proceedings of the 1992 ACM Computer Science Conference*, 407-414.

Carpenter, D.A. (1993.) Improving quality of teleprocessing line speed decisions. *Proceedings of 24th Annual Meeting of Midwest Decision Sciences Institute.*
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. (Apr 1986.) Microcomputer programs can be adapted for data network design, *Data Communications*, 116-128.
Held, G. (Oct 1983.) No more guesswork for sizing network components, *Data Communications*, 199-211.
Martin, J. (1972.) *Systems Analysis for Data Transmission*. Englewood Cliffs, NJ: Prentice-Hall, 413-480.
McGregor, P. (1974.) Effective use of data communications hardware, *Proceedings of National Computer Conference.*
Spragins, J.D., Hammond, J.L. & K. Paulikowski, K. (1991.) *Telecommunications: Protocol and Design*. Reading, MA: Addison-Wesley, 231-242.
Stamper, D.A. (1999.) *Business Data Communications*, 5th *Ed*. Redwood City, CA: Benjamin Cummings, 193-194.
Tanenbaum, A. (1981.) *Computer Networks.* Englewood Cliffs: Prentice-Hall, 56-67.