5-1991

Minimizing the knowledge requirements in a package sorting environment

Dana Susan Carney

To the Graduate Council:

I am submitting herewith a thesis written by Dana Susan Carney entitled "Minimizing the knowledge requirements in a package sorting environment." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Industrial Engineering.

C. Hal Aikens, Major Professor

We have read this thesis and recommend its acceptance:

John Snider, Wayne Claycombe

Accepted for the Council:
Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a thesis written by Dana Susan Carney entitled "Minimizing the Knowledge Requirements in a Package Sorting Environment." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Industrial Engineering.

Charles H. Aikens

C. Hal Aikens, Major Professor

We have read this thesis and recommend its acceptance:

John V. [illegible]

Wayne Claycombe

Accepted for the Council:

CW Minkel

Associate Vice Chancellor and Dean of The Graduate School

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Master's degree at The University of Tennessee, Knoxville, I agree that the Library shall make it available to borrowers under rules of the Library. Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of the source is made.

Permission for extensive quotation from or reproduction of this thesis may be granted by my major professor, or in his absence, by the Head of Interlibrary Services when, in the opinion of either, the proposed use of the material is for scholarly purposes. Any copying or use of the material in this thesis for financial gain shall not be allowed without my written permission.

Signature Dana S Carney

Date April 12, 1991

"MINIMIZING THE KNOWLEDGE REQUIREMENTS IN A PACKAGE SORTING ENVIRONMENT"

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Dana Susan Carney

May 1991

DEDICATION

This thesis is dedicated to my parents

Dr. Patrick Jerome Carney

and

Mrs. Nyla Katherine Carney

who have given me invaluable education opportunities

and have stressed the importance of education in

my future success.

ACKNOWLEDGMENTS

I would like to thank my major professor, Dr. Hal Aikens, for his guidance and patience. I would also like to thank the other committee members, Dr. John Snider and Dr. Wayne Claycombe, for their comments and assistance over the past two years. I would like to express my thanks to my best friend and future husband, John, for his understanding and patience during those times when there was no light at the end of anything. He encouraged me to stick with it and do it (or else). I would also like to thank John's parents, Fred and Nettie Mae Sherrod, for sharing their home and computer, as well as offering constant support when the program had no intention of working. I would like to thank the members of my family who have always inspired me to do my best: Debbie Carney (Nairobi, Kenya), Mike, Angie, and Erin Carney (Lake Charles, Louisiana), and Kevin and Catherine Carney (Nashville, Tennessee). I would like to thank United Parcel Service for giving me the opportunity to complete my graduate program. Finally, I would like to thank my mother and father, Nyla and Jerry Carney, for their support, love, and inspiration. They showed me how I can accomplish anything I set my mind to.

ABSTRACT

This research attempted to solve the human factors problem of minimizing the knowledge requirements in a package sorting environment by using operations research techniques. An integer programming model was developed to minimize the knowledge requirements for a package sorter by comparing and assigning 50 individual loads to 6 different outbound areas. The results of the model reduced the knowledge requirements significantly, and the effect of less knowledge requirements on the package sorter's mental workload was also considered.

It was concluded that the integer programming model was an effective tool in minimizing the knowledge requirements for a package sorter. The benefits of the model were presented, and future applications in larger facilities were recommended.

TABLE OF CONTENTS

The Information Theory Approach

LIST OF TABLES

CHAPTER I

INTRODUCTION

Scientists, especially mathematicians, have often been occupied with questions of optimization. An optimization problem can be defined as one that attempts to find the greatest numerical value (maximization) or least possible value (minimization) of some numerical or symbolic mathematical function (Cooper and Steinburg, 1974). As early as 100 B.C., Heron of Alexandria studied the optimization problem of light traveling between two points by the shortest path, and Euclid in 300 B.C. was associated with the problem of finding the shortest distance that could be drawn from a point to a line (Pike, 1986). However, the real impetus for the use of optimization theory came with World War II and the development of the digital computer. In the 1940s Dantzig recognized the mathematical structure of some military logistics problems and developed the Simplex Method of linear programming (Dantzig, 1963). Linear programming has since moved from an interesting mathematical topic to one of the most widely applied optimization procedures.

The ability to solve large sets of linear equations has followed closely the increasing capabilities of the digital computer and has permitted linear programming to be applied to numerous industrial problems. One particular industrial problem can be found in the transportation industry, but more specifically, in a package distribution facility.

The Package Distribution Facility - A Hub and Spoke Concept

The basic design of a package distribution facility can best be modeled by the airline industry's concept of the hub and spoke routing system. Prior to airline deregulation in the United States in 1978, airline carriers used what was referred to as a "linear" route system in which passengers were forced to travel through a path of many airports to arrive at a destination. However, after deregulation, the airlines chose hub cities that had central geographic locations to create a hub and spoke system (Figure 1). Any station on the system was then, at most, one stop away from all other stations (Oum and Tretheway, 1990).

One type of hub and spoke routing structure that is of particular interest is the East-West Directional Hub, which best models the package distribution facility (Figure 2). Stations north or south of the hub either are not served, or they are served in a separate North-South Directional Hub. For example, American Airlines operates predominantly east-west hub operations in Chicago and Dallas/Fort Worth, and north-south oriented hubs in Nashville and Raleigh/Durham (Oum and Tretheway, 1990.) Passengers in this facility would arrive at one terminal (or spoke), travel through the hub in the center, and then depart through a different terminal

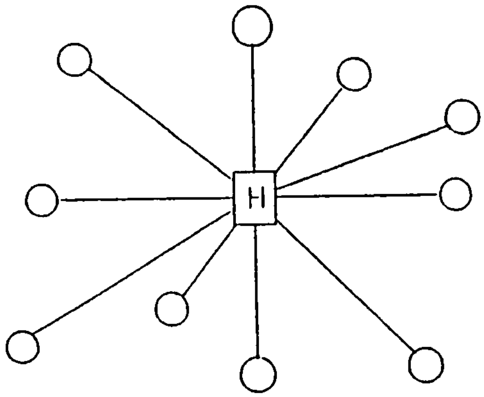


Figure 1: Airport Hub and Spoke System.

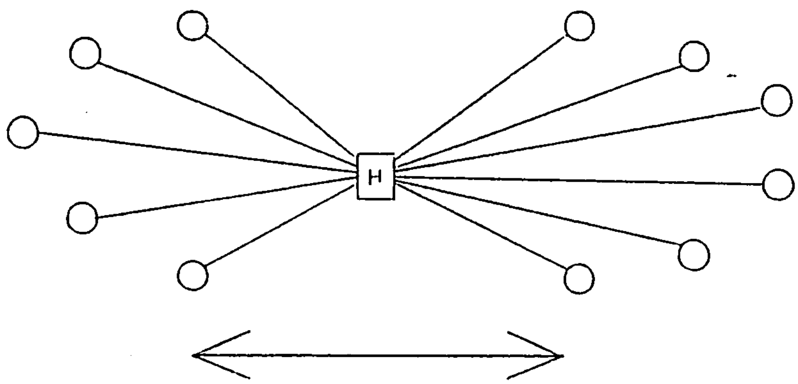


Figure 2: East-West Directional Hub.

(another spoke). The packages in a package distribution facility would behave similarly; however, the hub would then resemble a uni-directional facility in which certain terminals were considered arrival (or unload) terminals and the remaining terminals would be considered departure (or load) terminals (Figure 3). Once a package arrived (or was unloaded), it could not return to the arrival set of terminals.

Three particular goals must be accomplished in a package distribution facility: (a) unloading the packages, (b) sorting the packages to their proper outbound areas, and (c) loading the packages. As seen in Figure 4, a typical package distribution facility can have many outbound areas to which packages must be sorted.

In order for the sorter to know which packages should be sent to each outbound belt, each outbound destination must have a corresponding load chart that specifically shows which packages (listed by zip code, state, and city) will be allowed in each outbound load. To sort the packages, a sorter must first memorize the load chart for each outbound destination. Then the sorter can read each package's label to determine which one of the many destination loads that particular package must be sent, based on his ability to associate the zip code, state and city of the package with one of the load charts. After determining which destination load the package must be sent, the sorter must then recall which outbound area

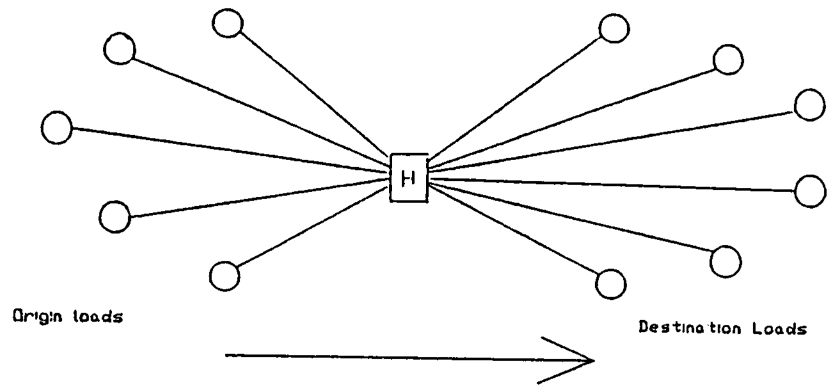


Figure 3: Uni-Directional Hub.

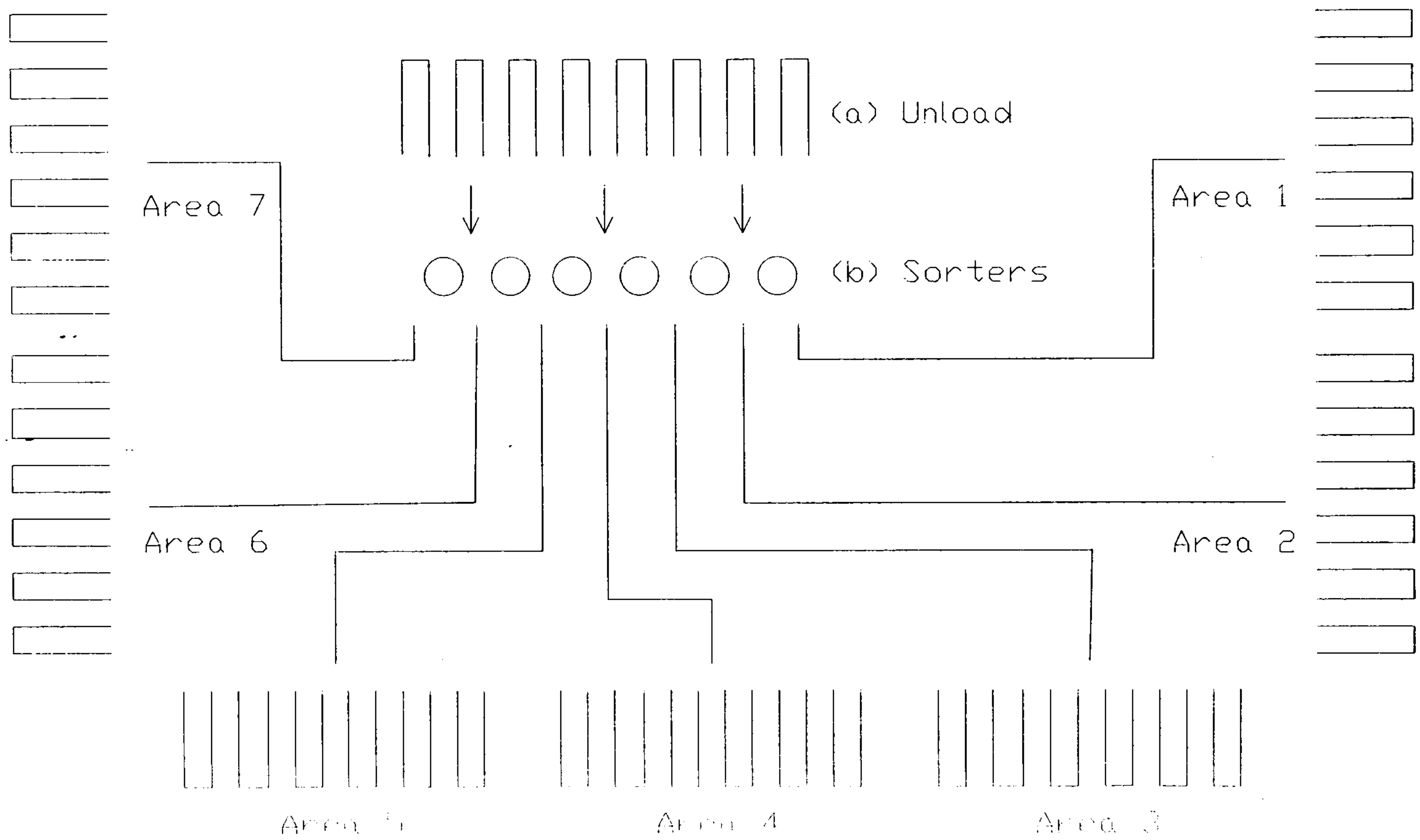


Figure 4: Package Distribution Facility.

that load is located on. Each package is then sorted to its proper outbound area, where it will be loaded in its correct outbound loads (see Areas 1-7 in Figure 4).

The Knowledge Unit

A knowledge unit is defined as one combination of:

(zip code, state, city)

or

(zip code range, state).

Therefore, as the number of outbound destinations (loads) increases, so does the number of knowledge units that a sorter must retain and be capable of recalling.

Two types of outbound destinations exist: hub loads and center loads. Hub loads are loads destined to other large package distribution (sorting) facilities. Hub loads mainly consist of zip code ranges as knowledge units (Table 1). Center loads are loads destined for delivery the next day, and they mainly consist of the more exact zip code, state, and city knowledge units (Table 2). Certain combinations of center or hub loads, when coupled together, will have less knowledge units than the sum of each individual centers' knowledge units (Table 3).

In larger package distribution facilities with a higher number of outbound areas and destination loads, the number of knowledge units that a sorter must be capable of retaining and recalling can be excessively high and should

Table 1: A Hub Load Chart, PA1.

State	Zip Codes
Pennsylvania	17000-19699

Table 2: A Center Load Chart, TN1.

State	Zip Codes
Athens	37303
Big Springs	37323
Calhoun	37309
Coker Creek	37314
Copperhill	37317
Decatur	37322
Delano	37325
Ducktown	37326
East Sweetwater	37874
Englewood	37329
Erie	37846
Etowah	37331
Farner	37333
Grandview	37337
Hiwassee College	37354
Isabella	37346
Loudon	37774
Madisonville	37354
Mount Vernon	37358
Niota	37826
Philadelphia	37846
Postelle	37368
Reliance	37369
Riceville	37370
Spring City	37381
Sweetwater	37874
Tellico Plains	37385
Ten Mile	37880
Turtletown	37391
Vonore	37885
Watts Bar Dam	37395

Table 3: Two hub load charts, PA1 and PA2, coupled together to create less knowledge units than the total of each individual chart's knowledge units.

State	Zip Codes	# Knowledge Units
PA1		
Pennsylvania	17000-19699	1
Total for PA1:		1
PA2		
Canada	All	1
New York	12800-14999	1
Pennsylvania	15000-16999	1
Total for PA2:		3
PA1 and PA2		
Canada	All	1
New York	12800-14999	1
Pennsylvania	15000-19699	1
Total for PA1 and PA2:		3

be minimized to reduce sorter training time and increase productivity. One approach to minimizing the number of knowledge units that a sorter must memorize and recall is to create a linear programming model which will consider each combination of hub and center loads and assign them to one of the many outbound areas. However, certain facility constraints must also be considered. For example, each of the many outbound areas can only hold a fixed, finite number of outbound destinations, and the cost of building a new facility would obviously not be justified for the sole purpose of reducing the number of knowledge units that a sorter must learn.

Another concern in operating the facility is the simulation of the facility to keep the packages "flowing" to the outbound areas at a constant rate so that the employees may remain in their immediate work area and allow the "work to come to them," thus reducing unproductive walk time between areas. Therefore, the number of packages sorted to each outbound area must be somewhat equal, putting yet another constraint on the linear programming model.

Calculating the Total Knowledge Units

The specific package distribution facility that will be modeled using a linear programming framework has 43 different inbound loads (loads to be unloaded), 6 separate outbound areas, and 56 distinct outbound loads. Each of the six outbound areas can have no more than 10 outbound destinations assigned to them.

To calculate the current number of knowledge units required to sort packages in this particular facility, an outbound line-up showing which loads are assigned to each outbound area of the current facility used in the model can be found in Table 4. Table 4 also shows the total number of knowledge units that each load chart currently has. Each of the load charts for all of the loads shown in Table 4 can be seen in Appendix I. Once the knowledge units for each individual load have been calculated, any combinations of knowledge units within each outbound area that would lessen the total can then be highlighted and recalculated (refer to Table 2).

The total number of knowledge units for the current outbound line-up in the facility is 783, which is the focus of the linear programming minimization problem.

Table 4: Current outbound line-up and the corresponding number of knowledge units contained in each load.

Loads	# Knowledge Units	Combinations
Area 1		
IL1	1	0
IL2	5	IL1+IL2 = 5
KY1	4	0
KY2	30	30
KY3	2	KY1+KY3 = 3
OH1	1	1
TN2	39	39
TN3	40	40
TN4	63	63
TN5	56	56
		TOTAL: 237
Area 2		
NC1	1	1
NJ1	2	2
OK1	1	1
SC1	1	1
TN6	37	37
TN7	12	12
VA1	3	0
VA2	3	VA1+VA2 = 5
VA3	3	VA1+VA3 = 2
		TOTAL: 61
Area 3		
FL1	2	2
MO1	8	8
MS1	2	2
NC2	80	0
NC3	49	0
NC4	68	0
NC5	47	NC2+NC3+NC4+NC5 = 1
TN8	32	32
TN9	35	35
		TOTAL: 80

Table 4: (cont.)

Loads	# Knowledge Units		Combinations
Area 4			
IN1	1		1
MA1	5		5
MD1	1		1
TN1	23		23
TN15	47		47
TN16	14		14
TN17	74		0
TN18	8	TN17+TN18=	2
TX1	2		2
		TOTAL:	95
Area 5			
GA1	64		0
GA2	79	GA1+GA2=	10
GA3	2		0
GA4	5	GA3+GA4=	4
TN10	68		0
TN11	65		65
TN12	28		28
TN13	109	TN10+TN13=	39
TN14	30		30
WI1	2		2
		TOTAL:	178
Area 6			
AR1	1		1
NJ2	1		1
OH2	1		1
PA1	1		0
PA2	3	PA1+PA2=	3
TN19	42		42
TN20	34		34
TN21	24		24
TN22	26		26
		TOTAL:	132
TOTAL Knowledge Units:			783

CHAPTER II

THE NEED FOR LESS KNOWLEDGE UNITS

Before a sorter can memorize and recall all of the knowledge units needed to perform his job, he first must have the capacity, as well as the motivation and desire, to learn. Any device for storing information, including memory, must have at least three facets. One facet is responsible for the input of information; in the case of memory, this process is usually termed learning or acquisition. The second is responsible for storing the information; this is the stage to which the term memory itself most frequently refers. Finally, one must have the means of accessing the information in the memory store; the terms retrieval and recall refer principally to this process (Baddeley, 1976).

The Evolution of Mental Workload

Workload is a construct that is used to account for the amount of effort required of an individual to maintain a certain level of task performance (Reid, 1985). The general concept of workload has its roots in the performance of physical work. However, with the influx of automation into the work place, the role of the human operator has changed from that of a physical laborer to primarily an information processor (Meshkati, 1985). This evolution has forced the ergonomic focus to switch from the measurement of physical workload to that of mental workload.

A few studies have tried to incorporate individual differences in the measurement of mental workload, especially in the area of subjective ratings. Hart, Childress, and Hauser addressed the issue that people differ in their ratings of workload for a given task. They suggest that these individual differences may be attributed to the fact that there is a wide range of interpretations for the term "workload" by operators (Hart, et al, 1982). Results showed that there seems to be several basic meanings to "workload" which were not at all similar, including time stress, emotional stress, task demands, cognitive effort, physical effort, motivation, and level of performance achieved. For example, if an operator dislikes time stress but enjoys cognitive problem solving, then asking the operator to rate 'load' will produce very different results for a given task than will be obtained from someone whose semantic framework is the reverse (Hart, et al, 1982).

The Role of Memory Retention and Recall on Mental Workload

A range of experiments from Ebbinghaus onward have shown that the length of a sequence of digits to be learned is critical, with longer sequences, as well as more sequences, taking disproportionately more time to learn (Baddeley, 1976). Therefore, the initial time that a sorter would spend memorizing the load charts, prior to ever performing the job, would increase as the number of knowledge units increased. Even after a sorter has memorized the knowledge units, his capacity to retain and recall them is variable, and the mental workload that the sorter must manage is quite complex. Also, the quality of the output from the sorters is affected by sorter-to-sorter differences in the capacity of each sorter to retain and accurately recall large numbers of knowledge units.

The complexity of mental workload has resulted in numerous proposed measurements by many different methods; however, poor reliability and lack of congruence among the different techniques for measuring mental workload (e.g., physiological, performance based, and subjective ratings) are major drawbacks to their practical application (Meshkati, 1985). Various writers (e.g., Leplat, 1978) maintain that mental workload should be tied to personality, task, physical, or physiological variables, and to such social variables as social pressure and expectations (Moray, 1982). Although data on subjective mental workload are astonishingly sparse, measurements of it may be conveniently divided into four groups, related to: (a) physical and physiological task parameters, (b) cognitive tasks, (c) manual control tasks, and (d) "time stress" (Hicks and Wierwille, 1979). Since a sorter in a package distribution facility has as a production standard the sorting of at least 1000 packages per hour, "time stress" may be the best measurement application.

Borg (1978a, 1978b) examined the implicit assumption that what is perceived difficult is considered to produce subjective mental workload; however, his experiments did not subject the participant to pressure associated with a continuous stream of signals that may arrive before the participant has finished dealing with an earlier signal, as in the case of a sorter facing a moving belt with packages. Borg's subjects were not under any kind of time stress, and no experiment to date has actually linked the perceived difficulty theory with the time stress factor.

With package sorting, an operator may receive signals which he or she must begin to process while still processing an earlier signal. The effect of multiple signals can lead to time stress. However, package sorting is not the only occupation in which time stress and subjective mental workload are concerned. For example, almost all reports on air traffic control refer to these problems as well. Remarkably few attempts have been made to measure the relationship between the two factors. Senders (1979) has gone so far as to assert that unless there is time stress in a task, there is by definition no subjective mental workload.

The Information Theory Approach

One approach to measuring mental workload is the use of information theory to determine if the operator is being requested to process more information than he or she is capable of processing. Its relevance to human factors is based on the fact that it provides for the measurement of information, and the unit of measurement is the bit. The bit (symbolized by the letter H) is the amount of information necessary to decide between two equally likely alternatives, and is derived with the following formula:

$$H = \log_2 n$$

where n is the number of equally probable alternatives. The information theory approach relies on the supposition that the human being has a limited capacity for processing information. If this limited capacity can be described in terms of bits, the current number of knowledge units can be converted to bits to determine if it exceeds the channel's capacity (McCormick et al, 1982).

Unfortunately, the information theory approach is not effective in showing why the number of knowledge units should be decreased because the current outbound lineup is successfully being used by sorters in the facility; these sorters have proven that the 783 knowledge units do not exceed the limited capacity of the human mind by the way that they effectively and accurately do their job. Thus, the information theory approach was dismissed.

The Modified Cooper and Harper Scale

A major limitation of most workload measures is that they are typically developed for a specific application (Reid, 1985). However, the most popular and widely accepted decision-tree rating scale for mental workload measurement, which has been used successfully to measure the mental workload of airline pilots, is the Cooper and Harper scale (Cooper and Harper, 1969). This scale, in its original form, was well suited for estimation of workload in manual control systems. Wierwille developed a modification of the scale, called the Modified Cooper-Harper (MCH) scale, which could be applied in mental workload estimation, regardless of the type of loading imposed by the task (Skipper, et al, 1986). The MCH scale can be seen in Appendix II.

The MCH scale has a 3-3-3-1 decision-tree scale structure. This scale uses the same decision-tree structure as the original Cooper-Harper scale; however, scale wording has been changed to increase the range of applicability and to place emphasis on mental workload. The MCH scale is not unidimensional in that it deals with performance, errors and workload. Furthermore, it contains a decision tree. Consequently, the possibility exists that using more categories might produce a more sensitive scale; however, experiments show that other scales using more categories do not possess as high a consistency and sensitivity as the MCH scale (Skipper, et al, 1986).

This particular scale would rate the overall task of sorting packages on a scale ranging from 1 to 10, with 1 being rated "very easy, highly desirable" and 10 being rated "impossible." Ideally, the desired rating for a particular job would be within the 1 (very easy, highly desirable) to 4 (minor but annoying difficulty) range on the MCH scale.

The most effective and accurate way to calculate a rating on the MCH scale for the current package sorting job is to have each sorter rate his job after performing it. Unfortunately, variation among the sorters with respect to experience, training, and attitude would lead to inconsistencies within the ratings. If an unacceptable rating was calculated (a rating of 5 or greater), the management group would then need to weigh the cost of implementing a new setup with less knowledge units versus the benefits. This new setup would be derived from the integer programming model. The costs associated with the new setup would consist of initial training time for all the sorters to learn the new setup as well as the loss of production directly following implementation.

Numerical Digit Recall

Regardless of the complexity of the sorter's mental workload, the problem of recalling the information associated with the load charts becomes an even greater concern. One experiment by Sternberg (1966) involved presenting the subject with a sequence of one to six digits, allowing 2 seconds for rehearsal, and then presenting a probe digit. The subject's task was to decide as quickly as possible whether or not the digit was in the presented sequence and to press a "yes" or "no" lever accordingly. Reaction time increased linearly with the number of digits presented. Sternberg suggests a model in which items in memory are scanned very rapidly using a comparator to determine whether a match exists between each item and the probe. If a match occurs, the subject responds "yes," and if none of the items matches the probe, he responds "no." The linear increase in reaction time with size of set can then be explained by assuming that items are scanned serially one at a time; hence, the greater the number of items, the longer the reaction time (Baddeley, 1976). This research supports an argument for reducing the number of knowledge units in the package sorting environment. An increase in operator productivity is an expectation as the time required to recall information and react to it decreases.

Training and Retention

One major factor affecting the ability of the sorter to retain and recall the required number of knowledge units is the initial training provided. An effective operator training program can be identified by at least two criteria. First, it provides trainees with the opportunity to develop knowledge and appropriate skills for safe and efficient system operation. Second, the knowledge and skills obtained in the training program are retained long enough for effective application to job performance. Techniques must be built into the training programs to minimize skills loss and provide retraining in areas where significant skills loss is likely to occur, as in the area of memorization and retention of load charts. Retention is especially difficult in contexts where there is a dissimilarity between the training and real-world environments, such as computer-based training in comparison to hands-on training (O'Hara, 1990). The demonstration by Von Restorff (1933) that an isolated item in an otherwise homogeneous list will be better recalled than a homogeneous item proved that the way the material was presented in training has some effect on how well the subject can retain and recall it. Thus, a three-digit number will be better learned if it presented within a list of nonsense syllables than if it is surrounded by other numbers (Baddeley, 1976). Therefore, the initial presentation of the outbound load charts that a sorter must memorize can be as important as the number of knowledge units that the sorter must learn.

Obviously, many different types of training programs should be experimented with to determine which one can achieve the greatest results with the least amount of time and money; however, working to reduce the required mental workload (e.g., knowledge units in the case of the package sorter), is probably one of the most beneficial areas in creating an effective and efficient training plan.

CHAPTER III

THE KNOWLEDGE UNIT MATRIX

The current number of knowledge units required for a sorter to sort packages at the current distribution facility (seen in Table 4) is 783. However, prior to formulating the integer programming problem to optimize (minimize) the number of knowledge units, a knowledge unit matrix was created to show the interaction between each of the 50 outbound destinations.

The Original Knowledge Unit Matrix

The original knowledge unit matrix that was constructed to solve this problem was actually a 50 by 50 by 6 matrix (or 15,000 elements). This particular matrix was a symmetric matrix across the xy axis in which each of the 50 (56 total loads - 6 sets of double loads = 50 total loads) outbound loads was compared with each other to show how the pairwise combinations would contribute to decreasing or increasing the number of total knowledge units. However, this 15,000 element matrix was too large to solve using the mainframe software currently available at the University of Tennessee, Knoxville.

To combat this problem, either the number of constraints or the number of variables had to be decreased. Many different attempts were made to decrease the number of constraints. For example, the original knowledge unit matrix was symmetric across the xy axis, and the constraints were decreased 50% by eliminating the lower half of the matrix by formulating only those constraints where x was greater than y. After many other similar attempts to decrease the number of constraints to a level that would not cause a "variable integer overflow," the knowledge unit matrix was simplified by placing half (25) of the outbound loads on one axis, and the remainder on the other axis. Of course, some knowledge of the loads had to be used in order to place pairs of loads that would decrease the number of knowledge units on opposite axis. A simple decision-making process was used to place each load on an opposite axis of the loads that would in any way decrease the total knowledge units. This decision-making process could eventually be modeled by an expert system in which a simple heuristic could be developed to reduce the knowledge unit matrix. The original knowledge unit matrix, along with the original formulation of the problem can be seen in its entirety in Appendix III.

<u>The Feasible Knowledge Unit Matrix</u>

A 25 by 25 (or 625 element), two dimensional matrix was then developed to show how each outbound load's knowledge units interacts with another outbound load's knowledge units. The number of belts represented the third dimension since constraints regarding the belt capacities must be adhered to.

For example, once this matrix is used for each of the 6 belts, it becomes a 25 by 25 by 6 matrix (or 3750 element). Thus, the number of outbound belts also contributes to the size of this problem. This specific matrix was designed to show how "the sum of two loads' knowledge units could actually be less than the total of the two individual loads' knowledge units." This matrix can be seen in Appendix IV.

For example, referencing the PA1 load, the total number of knowledge units assigned to that individual load is 1 (see PA1 in Table 4). Referencing the PA2 load, the total number of knowledge units assigned to that individual load is 3 (see PA2 in Table 4). However, when the PA1 load is coupled with the PA2 load, the total number of knowledge units is not 4, but 3 knowledge units! Thus, 1 + 3 does not equal 4 in this example because of the effect of combining the knowledge units in the two separate loads. This non-additive effect is true for many of the load combinations found in the knowledge unit matrix.

This matrix will, in effect, show the linear programming solver which two loads should be combined and assigned to the same outbound belt in order to assist in finding the minimum number of knowledge units required (hopefully less than the current 783). However, one drawback in using only a two dimensional matrix is that a problem solution is constrained to pairwise combinations. In some instances, assignments of three or four particular outbound loads will result in a decrease in the total number of knowledge units. For example, the TN1, TN5, TN15, and TN16 loads, when assigned individually to four separate outbound belts, have a total of 123 knowledge units; however, when all four loads are assigned to the same outbound belt, the total number of knowledge units required is reduced to 2. The two dimensional knowledge unit matrix can benefit from the assignment of all four of these loads if and only if two loads are combined together to create yet another load. For example, the TN1 and TN5 loads were combined together to create one load (referred to as the TN1 load). By combining key loads prior to formulating the remainder of the linear programming problem, the knowledge unit matrix can benefit even more from the combinations of loads that need to be assigned. Other loads that have been combined include: TN15 and TN16 (referred to as TN15), TN7 and TN9 (referred to as TN7), TN8 and TN11 (referred to as TN8), NC2 and NC3 (referred to as NC2), as well as NC4 and NC5 (referred to as NC4).

Total Package Matrix and Total Load Matrix

Two other matrices that were essential to the model are the total package matrix and the total load matrix (see Appendix V). There is no maximum limit to the number of knowledge units required to perform the package sorter's job; however, certain facility constraints limit the number of packages as well as the number of outbound destinations on one belt. A 25 by 25 by 6 (3750 element) matrix was created to show how many packages would be assigned to an outbound belt if any pairwise combination of loads was assigned to that belt. The maximum number of pieces on each outbound belt is 9500.

The second matrix, the total load matrix, was created to show how many total loads each pairwise combination would add if assigned to an outbound belt. Since preliminary combinations of loads were created to allow the model to account for combinations of three or four loads (for example, TN15 and TN16 are referred to as TN15), each of these new loads would actually represent 2 loads on an outbound belt instead of just 1. The maximum number of loads that can be assigned to an outbound belt is 10.

CHAPTER IV

THE LINEAR PROGRAMMING PROBLEM FORMULATION

The package distribution facility must complete three objectives: (a) unloading the packages, (b) sorting the packages to their outbound destinations, and (c) loading the packages. To sort the packages, a sorter must read each package's label and assign the package to one of K outbound (where K is a positive integer determined by the specifications of the facility). Each package is then sorted to its proper outbound belt where it will be sorted again to one of M outbound destinations (where M is a positive integer determined by facility constraints but less than or equal to N). A total of N possible outbound destinations, which will be determined by specifications of the facility, are assigned to K outbound belts. This particular configuration can be displayed as:

$$R \rightarrow B_k \rightarrow C_m \qquad (1.1)$$

where R = number of different inbound loads

B_k = k total outbound belts

C_m = m outbound destinations per belt

The Facility and Its Constraints

In order for the sorter to know which packages should be sent to each outbound belt, each outbound destination must have a corresponding load chart that will show which combinations of zip code, state, and city will be allowed in each outbound destination. A knowledge unit is defined as one combination of:

(zip code, state, city)

or

(zip code range, state)

Therefore, as the number of outbound destinations increases, so does the number of knowledge units that a sorter must retain.

In large package distribution facilities the number of knowledge units that a sorter must retain and recall has a negative impact on mental workload (and therefore productivity) as well as training requirements (refer to Chapter II). An objective to minimize the number of knowledge units would seem to be a logical choice. In this context one approach to minimizing this objective is to create a linear programming model which will consider each combination of hub and center loads and assign them to one of K belts. However, certain facility constraints must be considered as well. For example, each one of K outbound belts can only hold up to M outbound destinations (where M is a fixed, positive integer determined by facility specifications). The cost of building a new facility would obviously not be justified for the sole purpose of reducing the number of knowledge units that a sorter must learn.

Another concern in operating the facility is keeping the number of packages that are sorted to the outbound belts somewhat equal so that the employees may remain in their immediate work area in order to eliminate unproductive walk time between areas. Therefore, if the number of packages assigned to each outbound belt must be approximately equal, this creates yet another set of constraints for the linear programming model.

The specific package distribution facility that will be used to model the linear programming problem has 43 different inbound loads (R), 6 different outbound belts (K), and 56 separate outbound destinations (I). The six belts can have no more than 10 outbound destinations (M) and 9500 packages C(K) assigned to them.

The Integer Programming Formulation

The validity and value of many linear programming models would be improved markedly if one could restrict selected decision variables to integer values. Since about 1970, almost all linear programming solution procedures have been augmented with a capability which allows the user to restrict certain decision variables to integer values. The package sorter's knowledge unit problem is one example of a 0/1 integer (or assignment) programming problem, where certain programming codes assume that integer variables are restricted to values 0 or 1. The 0/1 integer variable is used to represent a go/no-go decision and will equal one when a load is assigned to a particular belt, and it will equal 0 when it is not assigned to a belt. Integer programs can be very difficult to solve, and as the number of integer variables is increased the solution time may increase dramatically.

The mainframe software that is used to formulate the integer programming problem is GAMS (General Algebraic Modeling System). GAMS is designed to make the construction and solution of large and complex mathematical programming models more straightforward for programmers and more comprehensible to users of models from other disciplines. These other disciplines may include the industrial engineering, accounting, transportation, or any other function that may benefit from mathematical programming but do not possess strong computer programming skills. Because GAMS can make concise algebraic statements of models in a language that is easily read by both modelers and computers, GAMS can substantially improve the productivity of modelers and expand the usefulness of mathematical programming applications in policy analysis and decision making. One positive feature of the GAMS compiler is that the constraints are written in summation notation, just as a modeler would formulate the program.

Once the program is compiled in GAMS, it is then submitted to another separate program, which actually solves the problem. Linear and mixed-integer models created with GAMS are solved with a special version of the ZOOM (Zero/One Optimization Methods) optimizer.

The formulation of the integer programming problem consists of the following:

$$Min \sum_i \sum_j \sum_k \; (B(i,j) * X(i,j,k)) \tag{1.2}$$

subject to

$$\sum_i \sum_j \; (A(i,j) * X(i,j,k)) \le C(k), \quad for\ all\ k \tag{1.3}$$

$$\sum_i \sum_j \; (T(i,j) * X(i,j,k)) \le L(k), \quad for\ all\ k \tag{1.4}$$

$$\sum_i \sum_k \; X(i,j,k) = 1, \quad for\ all\ j \tag{1.5}$$

$$\sum_j \sum_k \; X(i,j,k) = 1, \quad for\ all\ i \tag{1.6}$$

$$x(i,j,k) \in (0,1) \tag{1.7}$$

where B(i,j) = the number of knowledge units for load i coupled with load j, C(k) is the capacity of belt k in pieces, A(i,j) is the number of pieces associated with load i when it is coupled with load j, T(i,j) is the number of total loads when load i is coupled with load j, L(k) is the capacity in number of loads assigned to belt k, and X(i,j,k) is the integer variable that shows loads i and j are combined and are assigned to belt k.

Each of the constraints (equations 1.3 through 1.7) has a special purpose. The first constraint (equation 1.3) limits the capacity in packages to each belt. The second constraint (equation 1.4) limits the capacity in number of loads to each belt. The third and fourth constraints assign each load to only one belt (equations 1.5 and 1.6), and the fifth constraint allows all X(i,j,k) to equal 0 or 1, thus making this an integer programming problem.

Original Problems with Formulation

As stated in Chapter III, certain problems were encountered when the original form of the problem was submitted to the GAMS/ZOOM optimizer. The original formulation of the problem, seen in Appendix VI, was run using the GAMS/ZOOM software, but generated approximately 15,000 integer variables with over 6000 constraint equations to be solved. After the problem was compiled in GAMS, it was submitted to the ZOOM optimizer, which then could not solve the problem due to an "integer variable overflow." There were just too many variables and constraints for the software to solve. After many attempts to reduce the number of variables as well as constraints, the model was reconstructed and then solved.

The GAMS/ZOOM optimizer is the only software currently available on the VAX system at the University of Tennessee, Knoxville, that would have been capable of solving an integer problem of this magnitude. One concern should be noted about any future application of this model. This particular integer programming model formulates a problem for a relatively small package distribution facility with only 56 separate outbound loads, and the original model currently was too large to be solved using available software. If any further attempts are made to model facilities larger than the one modeled in this example, the reconstruction of the model, or rather the reconstruction of the knowledge unit matrix (as seen in Appendix III), would be a necessity.

CHAPTER V

THE SOLUTION TO THE INTEGER PROGRAMMING PROBLEM

Once the integer programming problem shown in Chapter IV was solved using the GAMS/ZOOM optimizer, the proposed feasible solution was verified and is shown in Table 5. All of the constraints were met, and the proposed solution calculated the optimum (minimized) number of total knowledge units equal to 423. This shows a reduction of 360 knowledge units, which is almost one half of the current 783 (54.02% of the original value).

Comparing the Current and the Proposed Outbound Lineups

Comparisons of the current outbound lineup with the proposed outbound lineup were made to determine the similarities and differences between the two, as well as considering implementation plans for the new lineup. Twenty-nine loads (or 51.79% of the total loads) that are coupled together in the proposed outbound lineup are on the same outbound belts with loads that they were coupled with on the current outbound lineup. This may suggest that the original lineup, which was created with the same initial constraints as the integer programming model, was probably a very good manual attempt to accomplish the same goals as the programming model. The number of unique combinations of

Table 5: Proposed outbound line-up with the corresponding number of knowledge units contained in each load.

Loads	# Knowledge Units		Combinations
<u>Area 1</u>			
GA4	5		5
IL1	5		0
IL2	5	IL1+IL2=	5
NJ2	1		1
TN2	39		0
TN4	63		63
TN14	30	TN2+TN14=	2
TN17	74		0
TN18	8	TN17+TN18=	2
TN19	42		42
		TOTAL:	120
<u>Area 2</u>			
FL1	2		2
KY1	4		4
MA1	5		5
NC1	1		1
NJ1	2		1
TN21	24		24
TN22	26		26
VA1	3		0
VA2	3		3
VA3	3	VA1+VA3 =	2
		TOTAL:	68
<u>Area 3</u>			
GA3	2		2
IN1	1		1
KY2	30		30
MD1	1		1
MO1	8		8
OK1	1		1
TN3	40		40
TX1	2		2
		TOTAL:	85

Table 5: (cont.)

Loads		# Knowledge Units		Combinations
Area 4				
GA1		64		0
GA2		79	GA1+GA2=	10
PA1		1		0
PA2		3	PA1+PA2=	3
TN1:	TN1	23		0
	TN5	62		0
TN10		68		0
TN13		109	TN10+TN13=	39
TN15:	TN15	47		0
	TN16	14	TN1+TN15=	2
			TOTAL:	54
Area 5				
AR1		1		1
MS1		2		2
TN6		37		37
TN7:	TN7			0
	TN9	47		0
TN8:	TN8			0
	TN11	97	TN7+TN8=	5
TN20		34		34
			TOTAL:	79
Area 6				
KY3		2		0
OH1		1		1
OH2		1		1
NC2:	NC2			0
	NC3	129		0
NC4:	NC4			0
	NC5	115	NC2+NC4=	1
SC1		1		1
TN12		28	KY3+TN12=	12
WI1		1		1
			TOTAL:	17
TOTAL Knowledge Units:				423

different loads on each outbound belt is too numerous to manually consider, especially when other constraints, such as the number of packages on each belt, must be included as well. The cost of manually determining the number of knowledge units in each of the many unique combinations would definitely be high; however, the integer programming model shown in Chapter IV is an excellent and cost effective tool that can be used to assist with this problem.

Implementation Considerations

Implementation plans for the proposed lineup need to be considered when changing from an existing lineup where 51.79% of the loads will remain paired together. Initial training costs associated with retraining all of the current employees coupled with lower production immediately following the implementation of the new lineup should be weighed against the decrease in knowledge units presented by the new outbound lineup. Of course, if the proposed lineup is going to be used in a new building with no current lineup, the initial training costs and low production following implementation would exist anyway and the proposed lineup would be most beneficial in this case.

CHAPTER VI

THE HUMAN FACTORS PROBLEM SOLVED WITH AN OPERATIONS RESEARCH APPROACH

The integer programming model shown in Chapter IV, along with its solution in Chapter V, are a good example of how a human factors problem can be solved using an operations research approach. The human factors problem, which concerns how much material an employee would have to memorize, retain, and recall, was solved using an integer programming model. So, who exactly is the benefactor when this problem is solved and implemented: the company or the employees? The answer to this question is that both the company and the employees actually benefit from the solution; however, each one benefits in a different way.

The Company's Benefits

The current computerized training and retention program used in this package distribution facility divides the total knowledge units into smaller units called decks. Each deck may contain up to 15 knowledge units. There are 5 different training programs that an employee must complete before being certified on a particular deck, these include: familiarization, study, drill, challenge, and certification. The employee must certify (achieve 100% in production, knowledge, and accuracy) on each deck before he is considered certified for the sorter's position. If each deck takes approximately 30 minutes to certify on, then the total certification time for a new employee would be determined by the total number of knowledge units that must be learned.

In the current outbound lineup containing 783 knowledge units (52 decks), a new employee would have to spend at least 1560 minutes (26 hours) to complete the computerized certification courses. However, with the proposed outbound lineup with 423 knowledge units (28 decks), a new employee would have to spend only 840 minutes (14 hours) completing the certification courses. If the company pays the employee an hourly rate of $9.00 per hour to complete the certification, it would see a decrease in training cost of $108.00 per employee (26-14 = 12 hrs x $9 = $108). However, if the company does not pay the employee the hourly rate while the employee completes the certification, it will still benefit from a reduction in the number of knowledge units. The benefits gained in this manner are less tangible, but the company will certainly benefit from having a less complicated and easier learned sort scheme. The time and effort that an employee will have to spend to learn it will decrease, thus making the job more appealing to those who wish to certify. The company will also see a reduction in errors, or sorting packages to the wrong destination, which in turn will save the number of rehandles in the building.

The Employee's Benefits

The company's benefits are important in the sense that the employer constantly desires to reduce training and labor costs in order to increase profit. However, too often the company's viewpoint is stressed, and the benefits that an employee would gain from new ideas are overlooked. In other words, if some cost savings or benefit to the company is not shown for implementing a new idea, the idea is conveniently overlooked. This is very unfortunate for the companies as well as the employees.

From the employee's viewpoint, learning a sort scheme would require much study time, as well as certification time that the company might not pay for; however, the long term benefits of having less knowledge units to retain would seem to outweigh this cost. Also, the employee who was trying to certify in order to earn more money as a sorter would actually invest less time initially memorizing and retaining the knowledge units, thus making the sorter's job more easily attainable. An opportunity cost is incurred by the employee prior to certifying; this opportunity cost can be defined as the differential increase that the employee would have received as a sorter for each hour that the employee was not certified. As the amount of time required to certify decreases, so does this opportunity cost.

Another consideration is the group of employees that might not have the mental capacity to retain 783 knowledge units, but would be capable of learning and performing the sorter's job with only 423. Thus, the opportunity cost that these employees have incurred would decrease as the employees certify for a position which they were not capable of achieving with more knowledge units.

A final consideration is that the package sorter's job with less knowledge units will cause a decrease in the mental workload that the sorter must face everyday on the job. The stress of having to retain and recall high numbers of knowledge units will lessen as well.

CHAPTER VII

DISCUSSION AND CONCLUSIONS

Although the problem of minimizing the number of knowledge units in a package distribution facility was solved, future research should be considered with respect to the following areas: the development of a specific measurement of mental workload in the case of the package sorter, the application of the integer programming problem to a larger facility, and post implementation results.

A Specific Measure of Mental Workload for the Package Sorter

The MCH scale was not specifically used to determine what type of rating (on a scale of 1 to 10, with a 10 representing an impossible setup) should be given to either the original outbound lineup or the proposed lineup with less knowledge units. This scale is rather vague in determining what kind of effect a large decrease in knowledge units would have on sorter as opposed to a slight decrease. In other words, the difference between a 4 and a 5 might be more distinguishable if the knowledge units decreased by 22 or 33 rather than 2 or 3. A specific numerical measure of mental workload that can be calculated by using the many variables (the number of knowledge units, the amount of signals the sorter receives in a specific time period, the number of outbound belts) would be desirable so that the effects of changing the number of knowledge units can immediately be seen.

Integer Programming Application in a Larger Facility

As shown in the original formulation of the integer programming problem (see Appendix VI), many problems were encountered with the variable integer overflow. The specific facility used to model the problem is actually a rather small, and some thought should be given to use the model (shown in Chapter IV) in a larger, more complicated facility for a few reasons. First, the application in a small facility was originally too large to formulate using the most powerful software available. Therefore, the problem was adjusted to stay within the bounds of the software. The effects of changing the knowledge unit matrix may be greater if the problem was formulated using a larger facility with more outbound loads. The total number of combinations of loads would increase, and any positive or negative effects of reducing the knowledge unit matrix may surface.

Post Implementation Results

Once the number of knowledge units have been minimized, actual implementation of the proposed lineup should be monitored for three reasons. First, any decrease in computerized training and retention time by the employees should be monitored to see if any correlation exists between the total decrease in knowledge units and the decrease in certification time. Second, a positive employee attitude can have a better effect on production, but may not be tangibly measured. The less time the employee has to invest in learning and certifying for a package sorter's job, the more positive his attitude will probably be. If the employees are responsible for retaining and recalling less knowledge units, the chance of making an error when sorting a package should also decrease. Thus, the employee is capable of doing a more effective job without the burden of remembering so many knowledge units. Finally, the cost of the implementation, if changing from a current lineup to a proposed lineup, should be compared to the benefits of the new lineup. The cost could be incurred as a decrease in production immediately following the implementation due to the confusion of the new lineup, as well as the cost of retraining and recertifying the employees on the new lineup. Of course, the benefits will have to be weighed on a different scale, because the production would eventually return to pre-implementation levels; thus, the implementation of a new lineup with less knowledge units would not necessarily dictate an increase in production.

Many other questions could be asked concerning the operations research approach to solving a human factors problem, but these three specific areas should be considered in future research to determine the appropriateness of using an integer programming model.

BIBLIOGRAPHY

BIBLIOGRAPHY

Baddeley, A.D. The Psychology of Memory, Basic Books, Inc., New York, N.Y., (1976), 16.

Borg, G. "Subjective Aspects of Physical Work." Ergonomics, 1978, 21, 215-220. (a)

Borg, G. "Subjective Effort in Relation to Physical Performance and Working Capacity." In Pick, H. (Ed.) Psychology from Research to Practice, New York: Plenum, 1978, 333-361. (b)

Cooper, G.E., and Harper, R.P., Jr. "The Use of Pilot Rating in the Evaluation of Aircraft Handling Qualities." Report TN-D-5153, National Aeronautics and Space Administration, Ames Research Center, Moffiet Field, California, 1969.

Cooper, L., and Steinberg, D. Methods and Applications of Linear Programming, W.B. Saunders Company, Philadelphia, P.A., (1974), 1.

Dantzig, G.B. Linear Programming and Extensions, Princeton University Press, Princeton, N.J. (1963).

Hicks, T., and Wierwille, W. "Comparison of Five Mental Workload Assessment Procedures in a Moving-Base Simulator." Human Factors, 1979, 21, 129-144.

Hurlack, R., and Montague, W. "Skill Retention and Its Implications for Navy Tasks: An Analytical Review." NPRDC SR 82-21, Navy Personnel Research and Development Center, San Diego, California, 1982.

Itoh, Y., and Hayashi, Y. "The Ergonomic Evaluation of Eye Movement and Mental Workload in Aircraft Pilots." Ergonomics, 1990, 33, 719-733.

Leplat, J. "Factors determing workload." Ergonomics, 1978, 21, 143-149.

Meshkati, N. "Human Mental Workload." Ergonomics International '85, London, 1985, 100-102.

McCormick, E., and Sanders, M. Human Factors in Engineering and Design, McGraw-Hill, Inc., (1982), 41-42.

Moray, N. "Subjective Mental Workload." Human Factors, Santa Monica, CA, (1982), 24(1), 25-40.

O'Hara, J. "The Retention of Skills Acquired Through Simulator-Based Training." Ergonomics, 1990, 33, 1143-1153.

Pike, R.W. Optimization for Engineering Systems, Van Nostrand Reinhold Company, Inc., New York, N.Y., (1986), 1.

Reid, G.B. "The Systematic Development of a Subjective Measure of Workload." Ergonomics International '85, London, 1985, 109-111.

Robertson, M., Hendrick, H. "Effect of Individual Differences and Perceived Difficulty in Coping with Mental Workload." Ergonomics International '85, London, 1985, 106-108.

Skipper, J., Rieger, C., Wierwille, W. "Evaluation of Decision-Tree Rating Scales for Mental Workload Estimation." Ergonomics, 1986, 29, 585-599.

Welford, A. "Mental Work-load as a Function of Demand, Capacity, Strategy and Skill." Ergonomics, 1978, Vol. 21, No. 3, 151-167.

Yeh, Y., and Wickens, C. "The Nature of Subjective Workload." Trends in Ergonomics/Human Factors II, North-Holland, Amsterdam, 1985, 153-160.

APPENDIXES

APPENDIX I

The GA2 Load Chart.

State	Zip Codes
Adairsville	30103
Alto Park	30161
Armuchee	30105
Beaumont	30736
Benedict	30125
Berryton	30748
Blackwood	30701
Burning Bush	30736
Cagle	30143
Carns Mill	30175
Cash	30701
Cassandra	30707
Cedar Grove	30707
Chamberlain	30728
Chelsea	30731
Cisco	30708
Cloudland	30709
Crandall	30711
Craneeater	30701
Callondale	30741
Davis Crossroads	30707
Desota Park	30161
Dyke	30540
Echota	30701
Eton	30724
Fairmont	30139
Fashion	30705
Fish Creek	30125
Foster Hills	30736
Friendship	30125
Gore	30747
Grady	30125
Guild	30728
Hassier Mill	30740
Hedrick	30710
Hillsdale	30728
Holland	30730
Huffaker	30161
Jasper	30143
Kensington	30728
Lafayette	30728
Lake Creek	30125
Lakeview	30741
Linwood	30145
Magby Gap	30752
Menlo	30731

The GA2 Load Chart: (cont.)

State	Zip Codes
Midway	30741
Naomi	30728
Nelson	30151
Noble	30728
Oremont	30125
Park City	30741
Pine Chapel	30701
Plainville	30733
Pond Spring	30707
Ramhurst	30705
Ranger	30734
Redbud	30701
Relay	30125
Resaca	30735
Rock City	30701
Rocky Face	30740
Rosedale	30701
Sallacoa	30139
Sherwood Forest	30161
Six Mile	30161
Spring Garden	30728
Sugartown	30755
Sumach	30705
Tate	30177
Trans	30728
Trickum	30755
Trion	30753
Tunnel Hill	30755
Varnell	30756
Walnut Grove	30728
Westside	30741
Whitestone	30186
Wooleys	30145

The GA4 Load Chart.

State	Zip Codes
Alabama	36000-36989
Georgia	30100-30289
Georgia	30400-30489
Georgia	30600-30689
Georgia	30800-31989

The IL2 Load Chart.

State	Zip Codes
Illinois	60800-61789
Montana	59000-59989
Oregon	97000-97989
Utah	84000-84729
Washington	98000-99489

The KY1 Load Chart.

State	Zip Codes
Kentucky	40200-40389
Kentucky	40500-40689
Kentucky	41000-41489
Ohio	45000-45899

The KY3 Load Chart.

State	Zip Codes
Albright	40419
Arkle	40734
Baldwin	40475
Berea	40403
Boneyville	40484
Buckeye	40444
Cane Creek	40739
Clover Bottom	40414
Crab Orchard	40419
Danville	40422
Fabert	40701
Foxtown	40432
Gatliff	40769
Bray hawk	40434
Bunns Chapel	40444
High Knob	40430
Indian Hills	40422
Junction City	40440
London	40741
Louden	40736
Moreland	40437
Nevisdale	40754
North Corbin	40701
Ottawa	40409
Parrot	40465
Portersburg	40765
Rowland	40484
Shelby City	40422
Walden	40768
Williamsburg	40769
Woodbine	40771

The MA1 Load Chart.

State	Zip Codes
Maine	04000-04999
Massachusetts	01000-02799
New Hampshire	03000-03999
Rhode Island	02800-02999
Vermont	05000-05999

The M01 Load Chart.

State	Zip Codes
Colorado	80000-81989
Iowa	50000-52899
Kansas	66000-67999
Missouri	63000-65899
Nebraska	68000-69989
North Dakota	58000-58989
South Dakota	57000-57889
Wyoming	82000-83189

The NC1 Load Chart.

State	Zip Codes
North Carolina	27000-28689

The AR1 Load Chart.

State	Zip Codes
Arkansas	71600-72999

The GA1 Load Chart.

State	Zip Codes
Aid	30521
Alto	30510
Aurarra	30534
Baldwin	30511
Belmont	30501
Blue Ridge	30513
Boydville	30577
Brooktown	30501
Bunker Hill	30512
Candler	30501
Cherry Log	30522
Chestoe	30501
Cleveland	30528
Colson Store	30535
Cornelia	30531
Crossroads	30516
Curtis	30513
Deercourt	30577
Demorest	30535
Dewey Rose	30634
Dicks Hill	30563
Eagle Grove	30520
Eastonollee	30538
Ellijay	30540
Epworth	30541
Fairview	30535
Fortsonia	30635
Gainesville	30501
Germany	30525
Habersham	30544
Hills	30523
Hulmeville	30635
Ivylog	30512
Klondike	30501
Leaf	30528
Lula	30554
Middleton	30635
Mize	30577
Morganton	30560
Mount Pleasant	30547
Mountain City	30562
Murrayville	30564
New Holland	30501
Nuberg	30634
Oakwood	30566
Porter Springs	30533

The GA1 Load Chart: (cont.)

State	Zip Codes
Red Hill	30557
Reed Creek	30643
Robertstown	30545
Satolah	30525
Saw Tooth	30552
Sells	30548
Silver City	30501
Suches	30572
Tiger	30576
Toccoa	30577
Tugalo	30577
Vandiver	30577
Westside	30501
Whitworth	30553
Wolffork	30568
Yahoola	30533
York	30568

The GA3 Load Chart.

State	Zip Codes
Georgia	30000-30099
Georgia	30300-30399

The IL1 Load Chart.

State	Zip Codes
Illinois	60000-60789

The IN1 Load Chart.

State	Zip Codes
Indiana	46000-47989

The KY2 Load Chart.

State	Zip Codes
Ages	40801
Alcolade	42511
Allock	41710
Altro	40863
Amuburgy	41801
Anco	41711
Arrowood	41712
Ary	40803
Asher	41713
Avawam	40804
Balkan	42657
Bandy	42501
Barnesburg	41753
Barridge	41753
Barrier	42633
Bath	41836
Bear Branch	41714
Bethesda	42633
Billows	42501
Bronston	42518
Cabell	42633
Calvin	40813
Chappell	40816
Cevrolet	40817
Clemons	41719
Cooper	42633
Dabney	42501
Delta	42613
Dixie	40849
Eadesville	42633
Engle	41741
Etna	42567
Faubush	42532
Flat Rock	42653
Frazier	42618
Gilreath	42635
Gray Knob	40829
Grundy	42501
Hall	41840
Harlan	40831
Hazard	41701
Hogue	42535
Kenvir	40847
Kodak	41773
Lamont	41340
Line Fork	41833

The KY2 Load Chart: (cont.)

State	Zip Codes
Lynch	40855
Mallie	42501
Monticello	42633
Napier	40859
Oak Hill	42501
Pearl	40863
Raven	41861
Ross Point	40806
Sandy Gap	42556
Smithboro	41759
Somerset	42501
Sugar Hill	42501
Tremont	40873
Valley Oaks	42501
Walnut Grove	42563
White Oak	42610

The MD1 Load Chart.

State	Zip Codes
Maryland	20600-21999

The MS1 Load Chart.

State	Zip Codes
Louisiana	70000-71499
Mississippi	38600-39789

The NC2 Load Chart.

State	Zip Codes
Almond	28702
Andrews	28901
Aquone	28703
Bell View	28906
Boiling Springs	28906
Brasstown	28902
Brendletown	28734
Briertown	28781
Burningtown	28734
Cane Creek	28906
Coalville	28901
Culberson	28903
East Franklin	28734
Ebenezer	28906
Ellijay	28734
Fontana Dam	28733
Franklin	28734
Gold Mine	28741
Grandview	28906
Grape Creek	28906
Hayesville	28904
Hewitt	28781
Hickory Knoll	28734
Higdonville	28734
Highlands	28741
Hiwassee Dam	28906
Holly Springs	28734
Hothouse	28903
Iotla	28734
Johnsonville	28903
Kyle	28781
Leatherman	28734
Macedonia	28903
Maltby	28905
Marble	28905
Martin Creek	28906
Mirrow Lake	28741
Mount Pleasant	28903
Murphy	28906
Nantahala	28781
Oak Grove	28734
Ogreeta	28906
Old Murphy	28906
Otto	28763
Panther Creek	28721
Peachtree	28906

The NC2 Load Chart: (cont.)

State	Zip Codes
Persimmon Creek	28906
Pinelog	28902
Prentiss	28734
Ranger	28906
Regal	28906
Rhodo	28901
Riverside	28734
Robbinsville	28771
Santeetlah	28771
Scaly Mountain	28775
Shookville	28741
Shooting Creek	28904
Short Off	28741
Slow Creek	28905
Stecoah	28771
Suit	28906
Sweet Gum	28771
Tallulah Gap	28771
Tapoco	28780
Thunderbird	28771
Tomotla	28905
Topton	28781
Tuskeegee	28771
Tusquitee	28904
Unaka	28908
Union	28734
Upper Peachtree	28906
Vests	28906
Violet	28908
Warne	28909
Watauga	28734
West Jutts Creek	28771
Wests Mill	28734
Yellow Creek	28771

A Hub Load Chart, PA2.

State	Zip Codes	
Canada	All	1
New York	12800-14999	1
Pennsylvania	15000-16999	1

The SC1 Load Chart.

State	Zip Codes
South Carolina	29000-29999

The TN10 Load Chart.

State	Zip Codes
Algood	38501
Allardt	38504
Allons	38541
Arnolds Chapel	38544
Asbury	38577
Bangham	38501
Banner Springs	38556
Baptist Ridge	38568
Baxter	38544
Ben Stockton	38556
Boatland	38566
Bonsack	38554
Butlers Landing	38551
Cedar Grove	38577
Celina	38551
Chestnut Mound	38552
Clark Range	38553
Coles Store	38544
Columbus Hill	38562
Cookeville	38501
Dale Hollow	38551
Davidson	38589
Double Springs	38544
Dudney Hill	38562
Enigma	38548
Ensor	38544
Fairview	38556
Freewill	38562
Gentry	38544
Goffton	38501
Granville	38564
Green Brier	38549
Grimsley	38565
Helena	38556
Holladay	38501
Independence	38573
Jamestown	38556
Jones Chapel	38549
Lancaster	38569
Laurelburg	38546
Littlecrab	38556
Martha Washington	38553
Martin Creek	38544
Moodyville	38549
Moss	38575
Nameless	38545

The TN10 Load Chart: (cont.)

State	Zip Codes
North Springs	38588
Oak Grove	38570
Oak Hill	38580
Oakley	38541
Pall Mall	38577
Parker	38577
Philadelphia	38545
Poplar Grove	38501
Red Hill	38549
Robbins	38549
Rocky Point	38501
Sadlers	38544
Shady Grove	38574
Shipley	38501
Stonewall	38567
Timothy	38568
Tinsleys Bottom	38551
Unity	38541
West Fork	38543
Wilder	38589
Windletown	38544
Wolf River	38577

The TN12 Load Chart.

State	Zip Codes
Alabama	35000-35999
Arizona	85000-86499
California	90000-96499
Florida	32400-32599
Illinois	61800-62999
Kentucky	40000-40199
Kentucky	42000-42489
Kentucky	42700-42789
Albany	42602
Alpha	42603
Bethelridge	42516
Catherine	42565
Clementsville	42539
Creston	42524
Dunnville	42528
Ingle	42536
Jamestown	42629
Liberty	42539
Mangum	42540
Middleburg	42541
Mintonville	42542
Russel Springs	42642
Sunnybrook	42650
Waterview	42650
Windsor	42565
Windy	42655
Yosemite	42566
Nevada	89000-89999

The TN14 Load Chart.

State	Zip Codes
Big Sandy	38221
Bruceton	38317
Buchanan	38222
Buena Vista	38318
Camden	38320
Clarksburg	38324
Como	38223
Cottage Grove	38224
Dresden	38225
Dukedom	38226
Eva	38333
Gleason	38229
Henry	38231
Holladay	38341
Hollow Rock	38342
Huntingdon	38344
Latham	38225
Leach	38349
Mansfield	38236
Martin	38237
McKenzie	38201
Palmersville	38241
Paris	38242
Puryear	38251
Sharon	38255
South Fulton	38257
Springville	38256
Vale	38317
Westport	38387
Yuma	38390

The TN16 Load Chart.

State	Zip Codes
Benton	37307
Birchwood	37308
Charleston	37310
Cleveland	37311
Conasauga	37316
East Cleveland	37311
Eureka	37311
Georgetown	37336
McDonald	37353
Ocoee	37361
Oldfort	37362
Parksville	37307
Tasso	37311
Wildwood Lake	37311

The TN18 Load Chart.

State	Zip Codes
Forest Hill	38138
Germantown	38138
Memphis	38128
Memphis	38133-38135
Memphis	38138
Raleigh	38134
Shelby Center	38128
Shelby Farms	38128

The TN2 Load Chart.

State	Zip Codes
Bargerton	38351
Bath Springs	38311
Beech Bluff	38313
Bemis	38314
Bonwood	38301
Chesterfield	38351
Darden	38328
Decaturville	38329
Denmark	38391
East Union	38301
Henderson	38340
Huntersville	38301
Huron	38345
Jacks Creek	38347
Jackson	38301
Juno	38351
Lexington	38351
Luray	38352
Magic Valley	38340
Malesus	38354
Medon	38356
Mercer	38392
Middlefork	38352
Montezuma	38340
Oakfield	38362
Parsons	38363
Pinson	38366
Providence	38301
Reagan	38368
Rose Hill	38301
Scotts Hill	38374
Shady Hill	38351
Spring Hill	38345
Springcreek	38378
Timberlake	38351
Uptonville	38392
Warrens Bluff	38351
Westover	38301
Wildersville	38388

The TN21 Load Chart.

State	Zip Codes
Adams	37010
Big Rock	37023
Bumpus Mills	37028
Burns	37029
Cedar Hill	37032
Charlotte	37036
Clarksville	37040
Colesburg	37055
Cumberland City	37050
Cumberland Furnace	37051
Cunningham	37052
Dickson	37055
Dover	37058
Erin	37061
Fredonia	37040
Henrietta	37015
Indian Mound	37079
Mulberry Hill	37058
Sailors Rest	37050
Sango	37040
Stayton	37051
Sylvia	37055
Tennessee City	37055
Thomasville	37015

The TN3 Load Chart.

State	Zip Codes
Ashland City	37015
Belle Meade	37205
Bellevue	37021
Bethpage	37022
Bordeaux	37218
Castalian Springs	37031
Chapmansboro	37035
Cottontown	37048
Crossplains	37049
East	37206
Fairview	37062
Gallatin	37066
Goodlettsville	37072
Greenbriar	37073
Hendersonville	37075
Inglewood	37216
Jere Baxter	37216
Joelton	37080
Kingston Springs	37082
Madison Springs	37082
Maplewood	37216
Millersville	37072
Nashville	37200-37203
Nashville	37205-37209
Nashville	37212-37213
Nashville	37216
Nashville	37218-37219
Nashville	37221
Nashville	37228
Nashville	37232
Nashville	37235-37240
Nashville	37244
Nashville	37246
Nashville	37250
New Deal	37048
Rockland	37075
Saundersville	37075
South Tunnel	37066
Uptown	37219
West	37209

The NC4 Load Chart.

State	Zip Codes
Asheville	28801-28802
Alexander	28701
Antioch	28753
Avery Creek	28704
Bakersville	28705
Ballantree	28803
Bandana	28705
Beaverdam	28715-28716
Big Laurel	28753
Blue Ridge	28711
Bluff	28743
Buckner	28754
Busick	28714
Candler	28715
Cane River	28714
Center Pigeon	28716
Cross Road	28731
Dellwood	28786
Dula Springs	28787
Emma	28806
Estatoe	28777
Fairview	28730
Flat Creek	28787
Foster Creek	28753
Gay	28779
Glady Fork	28715
Glenwood	28737
Grassy Creek	28777
Greens Creek	28779
Hamrick	28714
Happy Valley	28805
Hollifield	28752
Hyatt Creek	28786
Ivy	28754
Ivy Ridge	28754
Jacktown	28752
Juno	28748
Kemberly Woods	28804
Laurel	28753
Leicester	28748
Little Pinecreek	28753
Love Field	28779
Marion	28752
Micaville	28755
Mount Carmel	28706
New Candler	28715

The NC4 Load Chart: (cont.)

State	Zip Codes
North Cove	28752
Oak Forest	28703
Paint Fork	28709
Pensacola	28714
Pole Creek	28715
Red Hill	28705
Riceville	28805
Sand Hill	28806
Sherwood Forest	28778
Spruce Pine	28777
Stackhouse	28753
Sugar Hill	28752
Swill	28714
Tipton Hill	28740
Toledo	28740
Turnpike	28715
Venable	28803
Walnut	28753
West Canton	28716
Wing	28705
Woodfin	28804
Worley	28753

The NJ1 Load Chart.

State	Zip Codes
Connecticut	06000-06999
New York	10000-12799

The OH1 Load Chart.

State	Zip Codes
Ohio	43000-44989

The OK1 Load Chart.

State	Zip Codes
Oklahoma	73000-74999

A Hub Load Chart, PA1.

State	Zip Codes
Pennsylvania	17000-19699

A Center Load Chart, TN1.

State	Zip Codes
Athens	37303
Big Springs	37323
Calhoun	37309
Coker Creek	37314
Copperhill	37317
Decatur	37322
Delano	37325
Ducktown	37326
Englewood	37329
Etowah	37331
Farner	37333
Grandview	37337
Hiwassee College	37354
Isabella	37346
Madisonville	37354
Mount Vernon	37358
Postelle	37368
Reliance	37369
Riceville	37370
Spring City	37381
Tellico Plains	37385
Turtletown	37391
Watts Bar Dam	37395

The TN11 Load Chart.

State	Zip Codes
Alder Springs	37766
Andersonville	37705
Beech Grove	37769
Buckeye	37847
Caryville	37714
Cawood	37870
Chaska	37729
Clairfield	37715
Cotula	37729
Cove Creek	37714
Cumberland View	37757
Demory	37766
Disney	37769
Duff	37729
Eagan	37730
Elkmont	37738
Fincastle	37766
Flat Hollow	37870
Fraterville	37769
Gatlinburg	37738
Good Hope	37762
Grantsboro	37766
Highcliff	37762
Ivydell	37766
Jacksboro	37757
Jellico	37762
Kilsyth	37729
King	37715
Knapp	37769
Kodak	37764
Lafollette	37766
Lake City	37769
Little White Oak	37729
Marion	37715
Medford	37769
Meredith Cave	37766
Morley	37812
Newcomb	37819
Norris	37828
Oak Grove	37769
Oswego	37762
Pigeon Forge	37863
Pinecrest	37757
Pioneer	37847
Pittman Center	37738
Pruden	37851

The TN11 Load Chart: (cont.)

State	Zip Codes
Red Hill	37870
Royal Blue	37847
Russel Fork	37729
Sevierville	37862
Seymour	37865
Shea	37714
Silica	37714
Speedwell	37870
Stoney Fork	37714
Thackett Creek	37729
Turley	37714
Valley Creek	37715
Vasper	37714
Victory	37766
Welchs Camp	37714
Well Spring	37870
White Oak	37729
Wilkerson	37715
Wooldridge	37762

The TN13 Load Chart.

State	Zip Codes
Allens Chapel	37166
Amanda	38583
Bakers Crossroads	38555
Bates Hill	37110
Belk	37166
Belle Aire	38583
Bethany	37110
Big Lick	38555
Blue Hill	37110
Blue Springs	37166
Bluhmtown	37166
Board Valley	38583
Bon Air	38583
Bone Cave	38546
Bowman	38555
Bratcher	37110
Buckner	37166
Bybee	37110
Campaign	38550
Campbell Junction	38555
Cassville	38583
Centertown	37110
Clarktown	38583
Clifty	38583
Creston	38555
Crossville	38555
Cummingsville	38583
Daylight	37110
Dayton Spur	38555
De Rossett	38583
Dibrell	37110
Dodson	38583
Dorton	38555
Doyle	38559
Drop	38583
Eastland	38583
Erasmus	38555
Evins Mill	37166
Fairfield Glade	38555
Fairview	37110
Fanchers Mills	38583
Gath	37110
Goodbars	38581
Grassy Cove	37110
Homestead	38555
Howard Springs	38555

The TN13 Load Chart: (cont.)

State	Zip Codes
Hutchings	38583
Iboline	38555
Indian Mound	38583
Irving College	37110
Jefferson	37166
Jessie	37110
Johnsons Chapel	38583
Joppa	38587
Keltonburg	37166
Key	38583
Lake Tansi	38555
Lantana	38555
Laurel Cove	38585
Laurelburg	38546
Liberty	38595
Linary	38555
Lonewood	38585
Lost Creek	38583
Lucky	37110
Macedonia	38583
Mayland	38555
McElroy	38559
McMinnville	37110
Mechanicsville	37166
Mooneyham	38585
Mount Olive	37110
Mount Pisgah	38587
Mount Zion	37110
Mourberry	38583
Newton	38555
Nicholson Springs	37110
Northcutts Cove	37110
Oak Grove	38555
Oakdale	38583
Oakhill	38555
Oakland	37110
Peavine	38555
Peeled Chestnut	38583
Pine Grove	38585
Plateau	38555
Pomona Road	38555
Quebeck	38579
Ravenscroft	38583
Rinnie	38555
River Hill	38583
Riverview	38546

The TN13 Load Chart: (cont.)

State	Zip Codes
Rock Island	38581
Safely	37110
Shellsford	37110
Smithville	37166
Sparta	38583
Spencer	38585
Tabor	38555
Tarlton	37110
Volunteer Heights	38555
Walling	38587
Watkins	37166
Wayside	37110
Webbs Chapel	37166
White Hill	38546
Woodlawn	38555
Yager	37110
Yatestown	38587

The TN15 Load Chart.

State	Zip Codes
Alton Park	37410
Anderson	37376
Apison	37302
Bakewell	37304
Bobtown	37375
Brainerd	37411
Cagle	37327
Center Point	37327
Chattanooga	37400-37499
Coalmont	37313
Collegedale	37315
Comfort	37380
Dayton	37321
East Brainerd	37421
East Ridge	37412
Fairmont	37377
Graysville	37338
Build	37340
Harrison	37341
Hicks Chapel	37367
Hixson	37343
Jasper	37347
Kimball	37347
Lakesite	37379
Lewis Chapel	37327
Lookout Mountain	37350
Martins Springs	37380
Monteagle	37356
Mount Airy	37327
Nine Mile	37367
Ooltewah	37363
Pikeville	37367
Powells Crossroads	37397
Red Bank	37415
Richard City	37397
Saint Andrews	37372
Sampson	37367
Sherwood	37376
Signal Mountain	37377
Soddy Daisy	37379
Summerfield	37387
Tatesville	37365
Tracy City	37387
Whitwell	37397

The TN17 Load Chart.

State	Zip Codes
Almira	38011
Antioch	38058
Arlington	38002
Asbury	38069
Atoka	38004
Bailey	38017
Barretville	38053
Bartlett	38134
Beaver	38011
Bolton	38002
Braden	38010
Brighton	38011
Brunswick	38014
Charelston	38069
Clopton	38011
Cloverdale	38053
Collierville	38017
Cordova	38018
Crosstown	38004
Cuba	38053
Dancyville	38069
Dixonville	38053
Drummonds	38023
Eads	38028
East Acres	38053
Elba	38066
Ellendale	38029
Fisherville	38028
Forest Hill	38138
Gainsville	38049
Gallaway	38036
Germantown	38138
Hays	38057
Hickory Withe	38043
Holly Grove	38011
Hopewell	38011
Idaville	38004
Keeling	38069
Kerrville	38053
Kick	38017
Koko	38069
Lakeland	38002
Lambert	38068
Locke	38053
Longtown	38049
Lucy	38053

The TN17 Load Chart: (cont.)

State	Zip Codes
Mason	38049
Memphis	38128
Memphis	38134-38135
Memphis	38138
Millington	38054
Moscow	38057
Munford	38058
Oakland	38060
Piperton	38017
Pisgah	38018
Raleight	38134
Randolph	38004
Reverie	38062
Richardson	38004
Rossville	38066
Saint Paul	38004
Salem	38004
Shadowlawn	38002
Shelby Farms	38128
Somerville	38068
Tipton	38071
Union Hall	38004
Warren	38068
Wilkinsville	38053
Williston	38076
Woodstock	38053
Wright	38011
Yum Yum	38068

The TN19 Load Chart.

State	Zip Codes
Antioch	37013
Bairds Mill	37087
Bellwood	37087
Berry Hill	37204
Blair Lane	37087
Cairo Bend	37087
Cedar Bluff	37087
Centerville	37087
Creive Hall	37211
Doaks Crossroads	37087
Donelson	37214
Donelson	37229
Donelson	37231
Gladeville	37071
Greenwood	37087
Hermitage	37076
Hermitage Hill	37076
Hunters Point	37087
La Guardo	37087
Lavergne	37086
Lebanon	37087
Lebanon	37089
Leeville	37087
Linwood	37087
Major	37088
Martha	37087
Melrose	37204
Nashville	37076
Nashville	37078
Nashville	37087-37088
Nashville	37204
Nashville	37210-37211
Nashville	37214-37215
Nashville	37217
Nashville	37220
Nashville	37222
Nashville	37229-37230
Oakhill	37220
Taylorsville	37087
Tuckers Crossroads	37087
Vesta	37087
Vine	37087

The TN20 Load Chart.

State	Zip Codes
Allisona	37046
Almaville	37014
Arno	37146
Arrington	47014
Bending Chestnut	37064
Berrys Chapel	37064
Bethesda	37046
Bethlehem	37064
Beytonsville	37064
Bingham	37064
Boston	37064
Brentwood	37024
Brentwood	37027
Clovercroft	37046
College Grove	37046
Crosskeys	37046
Douglas	37064
Ewingville	37064
Fernvale	37064
Franklin	37064
Franklin	37065
Grassland	37064
Kingfield	37063
Kingfield	37064
Kirkland	37046
Leipers Fork	37064
Mallorys	37064
Paschall	37064
Peytonsville	37064
Reeds Store	37046
Riggs	37046
Rudderville	37064
Southall	37065
Triune	37014

The TN22 Load Chart.

State	Zip Codes
Athendale	38402
Belfast	37019
Bon Aqua	37025
Centerville	37033
Chapel Hill	37034
Coble	37033
Columbia	38401
Cornersville	37047
Culleoka	38451
Duck River	38454
Farmington	37091
Flatwoods	38458
Hampshire	38461
Hillsboro	37064
Hohenwald	38462
Kimmins	37081
Lewisburg	37091
Linden	37096
Littlelot	38454
Lobelville	37097
Lyles	37098
Mt Pleasant	38474
Primm Springs	38476
Sante Fe	38482
Williamsport	38487
Wrigley	37098

The TN4 Load Chart.

State	Zip Codes
Barfield	37130
Bel Aire	37130
Blackman	37130
Braxton	37190
Burt	37190
Center Hill	37190
Cherry Hill	37190
Commerce	37184
Compton	37130
Concord	37153
Cresent	37130
Culpepper	37149
Curlee	37190
Denver	37149
Dillton	37130
Donnels Chapel	37149
Eastside	37190
Florence	37130
Fruit Valley	37153
Greenvale	37184
Gum	37130
Halls Hill	37130
Hilltop	37167
Iconium	37190
Jakestown	37130
Jugtown	37130
Kittrel	37149
Leanna	37130
Leoni	37190
Little Hope	37130
Mankinville	37130
Milton	37118
Mona	37130
Mount Olive	37130
Mount Vernon	37153
Murfreesboro	37129-37133
Norene	37136
Overall	37130
Patterson	37153
Pleasant Ridge	37190
Pleasant View	37190
Porterfield	37118
Puckett	37153
Readyville	37149
Rock Springs	37167
Rockvale	37153

The TN4 Load Chart: (cont.)

State	Zip Codes
Rocky Fork	37167
Royer Estates	37130
Rucker	37130
Sharpsville	37130
Sheybogan	37190
Shiloh	37130
Shop Springs	37184
Silverhill	37130
Slatesville	37184
Smyrna	37167
Snell	37130
Statesville	37184
Veterans Admin	37130
Walter Hill	37184
Watertown	37184
Windrow	37153
Woodbury	37190

The TN5 Load Chart.

State	Zip Codes
Altamont	37301
Alto	37324
Arnold Air Force	37389
Beersheba Springs	37305
Belleville	37334
Belvidere	37306
Charity	37334
Cold Water	37334
Cowan	37318
Crisp Springs	37357
Decherd	37324
Elora	37328
Estill Springs	37330
Fairfield	37383
Fayetteville	37344
Flintville	37335
Harmony	37398
Harms	37334
Hillsboro	37342
Howell	37334
Hughey	37334
Huntland	37345
Kelso	37348
Lexie Crossroads	37306
Lynchburg	37352
Manchester	37355
Morrison	37357
Mulberry	37359
New Hope	37334
Normandy	37360
Park City	37344
Pelham	37366
Pleasant Grove	37160
Raus	37388
Skinem	37344
Summitville	37382
Tullahoma	37388
Viola	37394
Winchester	37398

The TN7 Load Chart.

State	Zip Codes
Alcoa	37701
Calderwood	37801
Friendsville	37737
Greenback	37742
Lenoir City	37771
Louisville	37777
Maryville	37801
Rockford	37853
Tallassee	37878
Top of the World	37878
Townsend	37882
Walland	37882

The TN9 Load Chart.

State	Zip Codes
Annadel	37770
Briceville	37710
Burrville	37712
Clinton	37716
Coalfield	37719
Corbin Hill	37840
Deer Lodge	37726
Devonia	37728
Dossett	37716
Edgemoor	37830
Elgin	37732
Emory Gap	37735
Fork Mountain	37728
Glenmary	37740
Harriman	37748
Helenwood	37755
Huntsville	37756
Kingston	37763
Midtown	37748
New River	37824
Oak Ridge	37830
Oakdale	37829
Oliver Springs	37840
Oneida	37841
Ozone	37842
Petros	37845
Robbins	37852
Rockwood	37845
Rosedale	37728
Rugby	37733
Stephens	37840
Sunbright	37872
Wartburg	37887
Westel	37889
Winfield	37892

The VA1 Load Chart.

State	Zip Codes
Virginia	23900-24189
Virginia	24400-24589
West Virginia	24700-26989

The VA3 Load Chart.

State	Zip Codes
Kentucky	41500-41689
Virginia	24200-24389
Virginia	24600-24699

The WI1 Load Chart.

State	Zip Codes
Minnesota	55000-56989
Wisconsin	53000-54999

The TN6 Load Chart.

State	Zip Codes
Biltomore	37643
Bloomingdale	37660
Blountville	37617
Blue Spring	37643
Braemer	37658
Bristol	37620
Butler	37640
Carter	37643
Central	37601
Church Hill	37642
Colonial Hgts	37663
Elizabethton	37643
Erwin	37650
Fall Branch	37656
Flagpond	37657
Fordtown	37663
Gray	37659
Hampton	37658
Johnson City	37601
Jonesboro	37659
Kingsport	37600
Limestone	37681
Lost Mountain	37681
Midfields	37665
Milligan College	37682
Morriston City	37660
Mount Carmel	37642
Orebank	37664
Piney Flats	37686
Rocky Forks	37643
Sadie	37643
Shell Creek	37687
Siam	37643
Springdale	37663
Telford	37690
Trade	37692
Valley Forge	37643

The TN8 Load Chart.

State	Zip Codes
Afton	37644
Bean Station	37708
Bulls Gap	37711
Bybee	37713
Centreville	37692
Chestnut Hill	37725
Chuckey	37643
Cosby	37722
Dandridge	37725
Del Rio	37727
Denton	37722
Edison	37731
Fry	37814
Hartford	37753
Jefferson City	37760
Kyles Ford	37765
Lowland	37778
Midway	37727
Midway	37809
Morristown	37814
Mosheim	37818
New Market	37820
Newport	37821
Rogersville	37857
Russelville	37860
Sneedville	37869
Surgoinsville	37873
Talbott	37877
Tate Springs	37708
Treadway	37883
White Pine	37890
Whitesburg	37891

The TX1 Load Chart.

State	Zip Codes
New Mexico	87000-88489
Texas	75000-79999

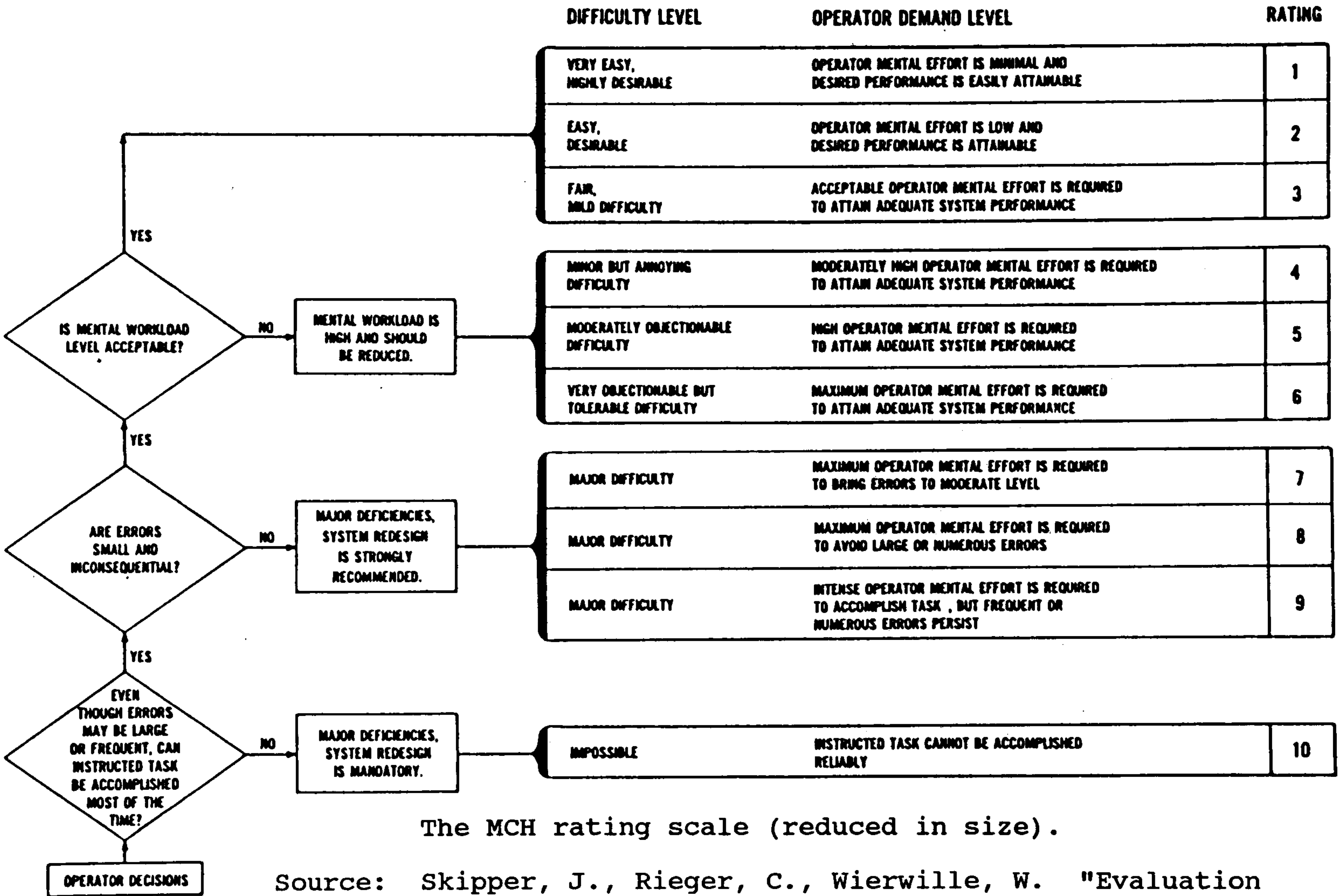
The VA2 Load Chart.

State	Zip Codes
Delaware	19700-19999
D.C.	20000-20599
Virginia	22000-23899

The FL1 Load Chart.

State	Zip Codes
Florida	32000-32399
Florida	32600-34999

APPENDIX II



The MCH rating scale (reduced in size).

Source: Skipper, J., Rieger, C., Wierwille, W. "Evaluation of Decision-Tree Rating Scales for Mental Workload Estimation." Ergonomics, 1986, 29, 585-599.

APPENDIX III

Original Knowledge Unit Matrix

To This Load:

From This Load	PA1	PA2	IL1	IL2	OK1	SC1	MO1	MS1	WI1	MA1	MD1	IN1	TX1	NJ2	OH2	AR1	FL1
PA1	1	3	2	6	2	2	9	3	3	6	2	2	3	2	2	2	3
PA2	3	3	4	8	4	4	11	5	5	8	4	4	5	4	4	4	5
IL1	2	4	1	5	2	2	12	3	3	6	2	2	3	2	2	2	3
IL2	6	8	5	5	6	6	13	7	7	10	6	6	7	6	6	6	7
OK1	2	4	2	6	1	2	9	3	3	6	2	2	3	2	2	2	3
SC1	2	4	2	6	2	1	9	3	3	6	2	2	3	2	2	2	3
MO1	9	11	12	13	9	9	8	10	10	13	9	9	10	9	9	9	10
MS1	3	5	3	7	3	3	10	2	4	7	3	3	4	3	3	3	4
WI1	3	5	3	7	3	3	10	4	2	7	3	3	4	3	3	3	4
MA1	6	8	6	10	6	6	13	7	7	5	6	6	7	6	6	6	7
MD1	2	4	2	6	2	2	9	3	3	6	1	2	3	2	2	2	3
IN1	2	4	2	6	2	2	9	3	3	6	2	1	3	2	2	2	3
TX1	3	5	3	7	3	3	10	4	4	7	3	3	2	3	3	3	4
NJ2	2	4	2	6	2	2	9	3	3	6	2	2	3	1	2	2	3
OH2	2	4	2	6	2	2	9	3	3	6	2	2	3	2	1	2	3
AR1	2	4	2	6	2	2	9	3	3	6	2	2	3	2	2	1	3
FL1	3	5	3	7	3	3	10	4	4	7	3	3	4	3	3	3	2
OH1	2	4	2	6	2	2	9	3	3	6	2	2	3	2	2	2	3
NJ1	3	5	3	7	3	3	10	4	4	7	3	3	4	3	3	3	4
KY1	5	7	5	9	5	5	12	6	6	9	5	5	6	5	5	5	6
TN2	40	42	40	44	40	40	47	41	41	44	40	40	41	40	40	40	41
TN3	41	43	41	45	41	41	48	42	42	45	41	41	42	41	41	41	42
TN4	64	66	64	68	64	64	71	65	65	68	64	64	65	64	64	64	65

Original Knowledge Unit Matrix (cont.)

To This Load:

From This Load	PA1	PA2	IL1	IL2	OK1	SC1	MO1	MS1	WI1	MA1	MD1	IN1	TX1	NJ2	OH2	AR1	FL1
TN1	63	65	63	67	63	63	70	64	64	67	63	63	64	63	63	53	64
TN15	62	64	62	66	62	62	69	63	63	66	62	62	63	62	62	62	63
TN13	110	112	110	114	110	110	117	111	111	114	110	110	111	110	110	110	111
TN10	69	71	69	73	69	69	76	70	70	73	69	69	70	69	69	69	70
KY2	31	33	31	35	31	31	38	32	32	35	31	31	32	31	31	31	32
KY3	3	5	3	7	3	3	10	4	4	7	3	3	4	3	3	3	4
TN12	29	31	29	33	29	29	36	30	30	33	29	29	30	29	29	29	30
NC1	2	4	2	6	2	2	9	3	3	6	2	2	3	2	2	2	3
VA1	4	6	4	8	4	4	11	5	5	8	4	4	5	4	4	4	5
TN6	38	40	38	42	38	38	45	39	39	42	38	38	39	38	38	38	39
VA2	4	6	4	8	4	4	11	5	5	8	4	4	5	4	4	4	5
TN7	48	50	48	52	48	48	55	49	49	52	48	48	49	48	48	48	49
TN8	98	100	98	102	98	98	105	99	99	102	98	98	99	98	98	98	99
VA3	4	6	4	8	4	4	11	5	5	8	4	4	5	4	4	4	5
NC2	130	132	130	134	130	130	137	131	131	134	130	130	131	130	130	130	131
NC4	116	118	116	120	116	116	123	117	117	120	116	116	117	116	116	116	117
GA1	65	67	65	69	65	65	72	66	66	69	65	65	66	65	65	65	66
GA2	80	82	80	84	80	80	87	81	81	84	80	80	81	80	80	80	81
GA3	3	5	3	7	3	3	10	4	4	7	3	3	4	3	3	3	4
GA4	6	8	6	10	6	6	13	9	9	10	6	6	7	6	6	6	7
TN17	75	77	75	79	75	75	82	76	76	79	75	75	76	75	75	75	76
TN18	9	11	9	13	9	9	16	10	10	13	9	9	10	9	9	9	10
TN19	43	45	43	47	43	43	50	44	44	47	43	43	44	43	43	43	44
TN20	35	37	35	39	35	35	42	36	36	39	35	35	36	35	35	35	36
TN21	25	27	25	29	25	25	32	26	26	29	25	25	26	25	25	25	26
TN22	27	29	27	31	27	27	34	28	28	31	27	27	28	27	27	27	28
TN14	31	33	31	35	31	31	38	32	32	35	31	31	32	31	31	31	32

Original Knowledge Unit Matrix (cont.)

To This Load:

From This Load	OH1	NJ1	KY1	TN2	TN3	TN4	TN1	TN15	TN13	TN10	KY2	KY3	TN12	NC1	VA1	TN6	VA2
PA1	2	3	5	40	41	64	63	62	110	69	31	3	29	2	4	38	4
PA2	4	5	7	42	43	66	65	64	112	71	33	5	31	4	6	40	6
IL1	2	3	5	40	41	64	63	62	110	69	31	3	29	2	4	38	4
IL2	6	7	9	44	45	68	67	66	114	73	35	7	33	6	8	42	8
OK1	2	3	5	40	41	64	63	62	110	69	31	3	29	2	4	38	4
SC1	2	3	5	40	41	64	63	62	110	69	31	3	29	2	4	38	4
MO1	9	10	12	47	48	71	70	69	117	76	38	10	36	9	11	45	11
MS1	3	4	6	41	42	65	64	63	111	70	32	4	30	3	5	39	5
WI1	3	4	6	41	42	65	64	63	111	70	32	4	30	3	5	39	5
MA1	6	7	9	44	45	68	67	66	114	73	35	7	33	6	8	42	8
MD1	2	3	5	40	41	64	63	62	110	69	31	3	29	2	4	38	4
IN1	2	3	5	40	41	64	63	62	110	69	31	3	29	2	4	38	4
TX1	3	4	6	41	42	65	64	63	111	70	32	4	30	3	5	39	5
NJ2	2	3	5	40	41	64	63	62	110	69	31	3	29	2	4	38	4
OH2	2	3	5	40	41	64	63	62	110	69	31	3	29	2	4	38	4
AR1	2	3	5	40	41	64	63	62	110	69	31	3	29	2	4	38	4
FL1	3	4	6	41	42	65	64	63	111	70	32	4	28	3	5	39	5
OH1	1	3	4	40	41	64	63	62	110	69	31	3	29	2	4	38	4
NJ1	3	2	6	41	42	65	64	63	111	70	32	4	30	3	5	39	5
KY1	4	6	4	43	44	67	66	65	113	72	34	3	31	5	7	41	7
TN2	40	41	43	39	79	102	101	100	148	107	69	41	67	40	42	76	42
TN3	41	42	44	79	40	103	102	101	149	108	70	42	68	41	43	77	43
TN4	64	65	67	102	103	63	125	124	73	131	93	65	91	64	66	100	66

Original Knowledge Unit Matrix (cont.)

To This Load:

From This Load	OH1	NJ1	KY1	TN2	TN3	TN4	TN1	TN15	TN13	TN10	KY2	KY3	TN12	NC1	VA1	TN6	VA2
TN1	63	64	66	101	102	125	62	2	171	130	92	64	90	63	65	99	65
TN15	62	63	65	100	101	124	2	61	170	129	91	63	89	62	64	98	64
TN13	110	111	113	148	149	73	171	170	109	39	139	111	137	110	112	146	112
TN10	69	70	72	107	108	131	130	129	39	68	98	70	96	69	71	105	71
KY2	31	32	34	69	70	93	92	91	139	98	30	31	58	31	33	67	33
KY3	3	4	3	41	42	65	64	63	111	70	31	2	12	3	5	39	5
TN12	29	30	31	57	68	91	90	89	137	96	58	12	28	29	31	65	31
NC1	2	3	5	40	41	64	63	62	110	69	31	3	29	1	4	38	4
VA1	4	5	7	42	43	66	65	64	112	71	33	5	31	4	3	40	5
TN6	38	39	41	76	77	100	99	98	146	105	67	39	65	38	40	37	40
VA2	4	5	7	42	43	66	65	64	112	71	33	5	31	4	5	40	3
TN7	48	49	51	86	87	110	109	108	156	115	77	49	75	48	50	84	50
TN8	98	99	101	116	117	160	159	158	206	165	17	99	125	98	100	95	100
VA3	4	5	6	42	43	66	65	64	112	71	33	5	31	4	2	40	6
NC2	130	131	133	168	169	192	191	190	238	197	159	131	157	130	132	166	132
NC4	116	117	119	154	155	178	177	176	224	183	145	117	143	116	118	152	118
GA1	65	66	68	103	104	127	126	125	173	132	94	66	92	65	67	101	67
GA2	80	81	83	118	119	142	141	140	188	147	109	81	107	80	82	116	82
GA3	3	4	6	41	42	65	64	63	111	70	32	4	30	3	5	39	5
GA4	6	7	9	44	45	68	67	66	114	73	35	7	33	6	8	42	8
TN17	75	76	78	113	114	137	136	135	183	142	104	76	102	75	77	111	77
TN18	9	10	12	47	48	71	70	69	117	76	38	10	36	9	11	45	11
TN19	43	44	46	81	82	105	104	103	151	110	72	44	70	43	45	79	45
TN20	35	36	38	73	74	97	96	95	142	102	64	36	62	35	37	71	37
TN21	25	26	28	63	64	87	86	85	132	92	54	26	52	25	27	61	27
TN22	27	28	30	65	66	89	88	87	134	94	56	28	54	27	29	63	29
TN14	31	32	34	2	70	93	92	91	139	98	60	32	58	31	33	67	33

Original Knowledge Unit Matrix (cont.)

To This Load:

From This Load	TN7	TN8	VA3	NC2	NC4	GA1	GA2	GA3	GA4	TN17	TN18	TN19	TN20	TN21	TN22	TN14
PA1	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
PA2	50	100	6	132	118	67	82	5	8	77	11	45	37	27	29	33
IL1	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
IL2	52	102	8	134	120	69	84	7	10	79	13	47	39	29	31	35
OK1	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
SC1	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
MO1	55	105	11	137	123	72	87	10	13	82	16	50	42	32	34	38
MS1	49	99	5	131	117	66	81	4	9	76	10	44	36	26	28	32
WI1	49	99	5	131	117	66	81	4	9	76	10	44	36	26	28	32
MA1	52	102	8	134	120	69	84	7	10	79	13	47	39	29	31	35
MD1	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
IN1	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
TX1	49	99	5	131	117	66	81	4	7	76	10	44	36	26	28	32
NJ2	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
OH2	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
AR1	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
FL1	49	99	5	131	117	66	81	4	7	76	10	44	36	26	28	32
OH1	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
NJ1	49	99	5	131	117	66	81	4	7	76	10	44	36	26	28	32
KY1	51	101	6	133	119	68	83	6	9	78	12	46	38	28	30	34
TN2	86	116	42	168	154	103	118	41	44	113	47	81	73	63	65	2
TN3	87	117	43	169	155	104	119	42	45	114	48	82	74	64	66	70
TN4	110	160	66	192	178	127	142	65	68	137	71	105	97	87	89	93

Original Knowledge Unit Matrix (cont.)

To This Load:

From This Load	TN7	TN8	VA3	NC2	NC4	GA1	GA2	GA3	GA4	TN17	TN18	TN19	TN20	TN21	TN22	TN14
TN1	109	159	65	191	177	126	141	64	67	136	70	104	96	86	88	92
TN15	108	158	64	190	176	125	140	63	66	135	69	103	95	85	87	91
TN13	156	206	112	238	224	173	188	111	114	183	117	151	143	133	135	139
TN10	115	165	71	197	183	132	147	70	73	142	76	110	102	92	94	98
KY2	77	127	33	159	145	94	109	32	35	104	38	72	64	54	56	60
KY3	49	99	5	131	117	66	81	4	7	76	10	44	36	26	28	32
TN12	75	125	31	157	143	92	107	30	33	102	36	70	62	52	54	58
NC1	48	98	4	130	116	65	80	3	6	75	9	43	35	25	27	31
VA1	50	100	2	132	118	67	82	5	8	77	11	45	37	27	29	33
TN6	84	95	40	166	152	101	116	39	42	111	45	79	71	61	63	67
VA2	50	100	6	132	118	67	82	5	8	77	11	45	37	27	29	33
TN7	47	5	50	176	162	111	126	49	52	121	55	89	81	71	73	77
TN8	5	97	100	226	212	161	176	99	102	171	105	139	131	121	123	107
VA3	50	100	3	132	118	67	82	5	8	77	11	45	37	27	29	33
NC2	176	226	132	129	1	193	208	131	134	203	137	171	163	153	155	159
NC4	162	212	118	1	115	179	194	117	120	189	123	157	149	139	141	145
GA1	111	161	67	193	179	64	10	66	69	138	72	106	98	88	90	94
GA2	126	176	82	208	194	10	79	81	59	153	87	121	113	103	105	109
GA3	49	99	5	131	117	66	81	2	4	76	10	44	36	26	28	32
GA4	52	102	8	134	120	69	59	4	5	79	13	47	39	29	31	35
TN17	121	171	77	203	189	138	153	76	79	74	2	116	108	98	100	104
TN18	55	105	11	137	123	72	87	10	13	2	8	50	42	32	34	38
TN19	89	139	45	171	157	106	121	44	47	116	50	42	76	66	68	72
TN20	81	131	37	163	149	98	113	36	39	108	42	76	34	58	60	64
TN21	71	121	27	153	139	88	103	26	29	98	32	66	58	24	50	54
TN22	73	123	29	155	141	90	105	28	31	100	34	68	60	50	26	56
TN14	77	107	33	159	145	94	109	32	35	104	38	72	64	54	56	30

The original formulation of the integer programming problem consisted of the following:

$$Min \sum_{i}\sum_{j}\sum_{k} (B(i,j) * X(i,j,k)) \quad (2.1)$$

subject to

$$\sum_{i}\sum_{j} (A(i) * X(i,j,k)) \leq C(k), \text{ for all } k \quad (2.2)$$

$$\sum_{i}\sum_{j} X(i,j,k) \leq L(k), \text{ for all } k \quad (2.3)$$

$$\sum_{i}\sum_{k} X(i,j,k) = 1, \text{ for all } j \quad (2.4)$$

$$\sum_{j}\sum_{k} X(i,j,k) = 1, \text{ for all } i \quad (2.5)$$

$$X(i,j,k) - X(j,i,k) = 0, \text{ for all } i,j,k \quad (2.6)$$

$$\sum_{i}\sum_{j} X(i,j,k) = 0, \text{ for all } i=j \quad (2.7)$$

where B(i,j) = the number of knowledge units for load i coupled with load j, C(k) is the capacity of belt k in pieces, A(i) is the number of pieces associated with load i, L(k) is the capacity in number of loads assigned to belt k, and X(i,j,k) is the integer variable that shows load i and j are combined and are assigned to belt k.

Each of the constraints (equations 2.2 through 2.7) has a special purpose. The first constraint (equation 2.2) limits the capacity in packages to each belt. The second constraint (equation 2.3) limits the capacity in number of loads to each belt. The third and fourth constraints assign each load to only one belt (equations 2.4 and 2.5). The fifth constraint makes sure there are symmetric assignments across the xy axis for each load. The sixth constraint (equation 2.7) does not allow assignment across the xy axis, so that each load will be forced into an assignment with another load.

APPENDIX IV

Knowledge Unit Matrix

To This Load:

From	PA2	IL2	SC1	MS1	MA1	IN1	NJ2	AR1	OH1	KY1	TN3	TN1	TN13
PA1	3	6	2	3	6	2	2	2	2	5	41	63	110
IL1	4	5	2	3	6	2	2	2	2	5	41	63	110
OK1	4	6	2	3	6	2	2	2	2	5	41	63	110
MO1	11	13	9	10	13	9	9	9	9	12	48	70	117
WI1	5	7	3	4	7	3	3	3	3	6	42	64	111
MD1	4	6	2	3	6	2	2	2	2	5	41	63	110
TX1	5	7	3	4	7	3	3	3	3	6	42	64	111
OH2	4	6	2	3	6	2	2	2	2	5	41	63	110
FL1	5	7	3	4	7	3	3	3	3	6	42	64	111
NJ1	5	7	3	4	7	3	3	3	3	6	42	64	111
TN2	42	44	40	41	44	40	40	40	40	43	79	101	148
TN4	66	68	64	65	68	64	64	64	64	67	103	125	73
TN15	64	66	62	63	66	62	62	62	62	65	101	2	170
TN10	71	73	69	70	73	69	69	69	69	72	108	130	39
KY3	5	7	3	4	7	3	3	3	3	3	42	64	111
NC1	4	6	2	3	6	2	2	2	2	5	41	63	110
TN6	40	42	38	39	42	38	38	38	38	41	77	99	146
TN7	50	52	48	49	52	48	48	48	48	51	87	109	156
VA3	6	8	4	5	8	4	4	4	4	6	43	65	112
NC4	118	120	116	117	120	116	116	116	116	119	155	177	224
GA2	82	84	80	81	84	80	80	80	80	83	119	141	188
GA4	8	10	6	9	10	6	6	6	6	9	45	67	114
TN18	11	13	9	10	13	9	9	9	9	12	48	70	117
TN20	37	39	35	36	39	35	35	35	35	38	74	96	142
TN22	29	31	27	28	31	27	27	27	27	30	66	88	134

Knowledge Unit Matrix (cont.)

To This Load:

From	KY2	TN12	VA1	VA2	TN8	NC2	GA1	GA3	TN17	TN19	TN21	TN14
PA1	31	29	4	4	98	130	65	3	75	43	25	31
IL1	31	29	4	4	98	130	65	3	75	43	25	31
OK1	31	29	4	4	98	130	65	3	75	43	25	31
MO1	38	36	11	11	105	137	72	10	82	50	32	38
WI1	32	30	5	5	99	131	66	4	76	44	26	32
MD1	31	29	4	4	98	130	65	3	75	43	25	31
TX1	32	30	5	5	99	131	66	4	76	44	26	32
OH2	31	29	4	4	98	130	65	3	75	43	25	31
FL1	32	28	5	5	99	131	66	4	76	44	26	32
NJ1	32	30	5	5	99	131	66	4	76	44	26	32
TN2	69	67	42	42	116	168	103	41	113	81	63	2
TN4	93	91	66	66	160	192	127	65	137	105	87	93
TN15	91	89	64	64	158	190	125	63	135	103	85	91
TN10	98	96	71	71	165	197	132	70	142	110	92	98
KY3	31	12	5	5	99	131	66	4	76	44	26	32
NC1	31	29	4	4	98	130	65	3	75	43	25	31
TN6	67	65	40	40	95	166	101	39	111	79	61	67
TN7	77	75	50	50	5	176	111	49	121	89	71	77
VA3	33	31	2	6	100	132	67	5	77	45	27	33
NC4	145	143	118	118	212	1	179	117	189	157	139	145
GA2	109	107	82	82	176	208	10	81	153	121	103	109
GA4	35	33	8	8	102	134	69	4	79	47	29	35
TN18	38	36	11	11	105	137	72	10	2	50	32	38
TN20	64	62	37	37	131	163	98	36	108	76	58	64
TN22	56	54	29	29	123	155	90	28	100	68	50	56

APPENDIX V

Total Package Matrix

To This Load:

From	PA2	IL2	SC1	MS1	MA1	IN1	NJ2	AR1	OH1	KY1	TN3	TN1	TN13
PA1	984	992	1133	992	813	980	847	1482	1191	875	3980	1980	966
IL1	959	967	1108	967	788	955	822	1457	1166	850	3955	1955	941
OK1	1523	1531	1672	1531	1352	1519	1386	2021	1730	1414	4519	2519	1505
MO1	1743	1751	1892	1751	1572	1739	1606	2241	1950	1634	4739	2739	1725
WI1	1299	1307	1448	1307	1128	1295	1162	1797	1506	1190	4295	2295	1281
MD1	928	936	1077	936	757	924	791	1426	1135	819	3924	1924	910
TX1	1500	1508	1649	1508	1329	1496	1363	1998	1707	1391	4496	2496	1482
OH2	1199	1207	1348	1207	1028	1195	1062	1697	1406	1090	4195	2195	1181
FL1	1282	1290	1431	1290	1111	1278	1145	1780	1489	1173	4278	2278	1264
NJ1	1114	1122	1263	1122	943	1110	977	1612	1321	1005	4110	2110	1096
TN2	1056	1064	1205	1064	885	1052	919	1554	1263	947	4052	2052	1038
TN4	1505	1513	1654	1513	1334	1501	1368	2003	1712	1396	4501	2501	1487
TN15	3716	3724	3865	3724	3545	3712	3579	4214	3923	3607	6712	4712	3698
TN10	1094	1102	1243	1102	923	1090	957	1592	1301	985	4090	2090	1076
KY3	1082	1090	1231	1090	911	1078	945	1580	1289	973	4078	2078	1064
NC1	1081	1089	1230	1089	910	1077	944	1579	1288	972	4077	2077	1063
TN6	2976	2984	3125	2984	2805	2972	2839	3474	3183	2867	5972	3972	2958
TN7	2529	2537	2678	2537	2358	2525	2392	3027	2736	2420	5525	3525	2511
VA3	1307	1315	1456	1315	1136	1303	1170	1805	1514	1198	4303	2303	1289
NC4	965	973	1114	973	794	961	828	1463	1172	856	3961	1961	947
GA2	676	684	825	684	505	672	539	1174	883	567	3672	1672	658
GA4	1228	1236	1377	1236	1057	1224	1091	1726	1435	1119	4224	2224	1210
TN18	563	571	712	571	392	559	426	1061	770	454	3559	1559	545
TN20	1612	1620	1761	1620	1441	1608	1475	2110	1819	1503	4608	2608	1594
TN22	1250	1258	1399	1258	1079	1246	1113	1748	1457	1141	4246	2246	1232

Total Package Matrix (cont.)

To This Load:

From	KY2	TN12	VA1	VA2	TN8	NC2	GA1	GA3	TN17	TN19	TN21	TN14
PA1	756	3884	1003	803	2433	777	589	1256	1309	3109	1121	1033
IL1	731	3859	978	778	2408	752	564	1231	1284	3084	1096	1008
OK1	1295	4423	1542	1342	2972	1316	1128	1795	1848	3648	1660	1572
MO1	1515	4643	1762	1562	3192	1536	1348	2015	2068	3868	1880	1792
WI1	1071	4199	1318	1118	2748	1092	904	1571	1624	3424	1436	1348
MD1	700	3828	947	747	2377	721	533	1200	1253	3053	1065	977
TX1	1272	4400	1519	1319	2949	1293	1105	1772	1825	3625	1637	1549
OH2	971	4099	1218	1018	2648	992	804	1471	1524	3324	1336	1248
FL1	1054	4182	1301	1101	2731	1075	887	1554	1607	3407	1419	1331
NJ1	886	4014	1133	933	2563	907	719	1386	1439	3239	1251	1163
TN2	828	3956	1075	875	2505	849	661	1328	1381	3181	1193	1105
TN4	1277	4405	1524	1324	2954	1298	1110	1777	1830	3630	1642	1554
TN15	3488	6616	3735	3535	5165	3509	3321	3988	4041	5841	3853	3765
TN10	866	3994	1113	913	2543	887	699	1366	1419	3219	1231	1143
KY3	854	3982	1101	901	2531	875	687	1354	1407	3207	1219	1131
NC1	853	3981	1100	900	2530	874	686	1353	1406	3206	1218	1130
TN6	2748	5876	2995	2795	4425	2769	2581	3248	3301	5101	3113	3025
TN7	2301	5429	2548	2348	3978	2322	2134	2801	2854	4654	2666	2578
VA3	1079	4207	1326	1126	2756	1100	912	1579	1632	3432	1444	1356
NC4	737	3865	984	784	2414	758	570	1237	1290	3090	1102	1014
GA2	448	3576	695	495	2125	469	281	948	1001	2801	813	725
GA4	1000	4128	1247	1047	2677	1021	833	1500	1553	3353	1365	1277
TN18	335	3463	582	382	2012	356	168	835	888	2688	700	612
TN20	1384	4512	1631	1431	3061	1405	1217	1884	1937	3737	1749	1661
TN22	1022	4150	1269	1069	2699	1043	855	1522	1575	3375	1387	1299

Total Load Matrix

To This Load:

From	PA2	IL2	SC1	MS1	MA1	IN1	NJ2	AR1	OH1	KY1	TN3	TN1	TN13
PA1	2	2	2	2	2	2	2	2	2	2	2	3	2
IL1	2	2	2	2	2	2	2	2	2	2	2	3	2
OK1	2	2	2	2	2	2	2	2	2	2	2	3	2
MO1	2	2	2	2	2	2	2	2	2	2	2	3	2
WI1	2	2	2	2	2	2	2	2	2	2	2	3	2
MD1	2	2	2	2	2	2	2	2	2	2	2	3	2
TX1	2	2	2	2	2	2	2	2	2	2	2	3	2
OH2	2	2	2	2	2	2	2	2	2	2	2	3	2
FL1	2	2	2	2	2	2	2	2	2	2	2	3	2
NJ1	2	2	2	2	2	2	2	2	2	2	2	3	2
TN2	2	2	2	2	2	2	2	2	2	2	2	3	2
TN4	2	2	2	2	2	2	2	2	2	2	2	3	2
TN15	3	3	3	3	3	3	3	3	3	3	3	4	3
TN10	2	2	2	2	2	2	2	2	2	2	2	3	2
KY3	2	2	2	2	2	2	2	2	2	2	2	3	2
NC1	2	2	2	2	2	2	2	2	2	2	2	3	2
TN6	2	2	2	2	2	2	2	2	2	2	2	3	2
TN7	3	3	3	3	3	3	3	3	3	3	3	4	3
VA3	2	2	2	2	2	2	2	2	2	2	2	3	2
NC4	3	3	3	3	3	3	3	3	3	3	3	4	3
GA2	2	2	2	2	2	2	2	2	2	2	2	3	2
GA4	2	2	2	2	2	2	2	2	2	2	2	3	2
TN18	2	2	2	2	2	2	2	2	2	2	2	3	2
TN20	2	2	2	2	2	2	2	2	2	2	2	3	2
TN22	2	2	2	2	2	2	2	2	2	2	2	3	2

Total Load Matrix (cont.)

To This Load:

From	KY2	TN12	VA1	VA2	TN8	NC2	GA1	GA3	TN17	TN19	TN21	TN14
PA1	2	2	2	2	3	3	2	2	2	2	2	2
IL1	2	2	2	2	3	3	2	2	2	2	2	2
OK1	2	2	2	2	3	3	2	2	2	2	2	2
MO1	2	2	2	2	3	3	2	2	2	2	2	2
WI1	2	2	2	2	3	3	2	2	2	2	2	2
MD1	2	2	2	2	3	3	2	2	2	2	2	2
TX1	2	2	2	2	3	3	2	2	2	2	2	2
OH2	2	2	2	2	3	3	2	2	2	2	2	2
FL1	2	2	2	2	3	3	2	2	2	2	2	2
NJ1	2	2	2	2	3	3	2	2	2	2	2	2
TN2	2	2	2	2	3	3	2	2	2	2	2	2
TN4	2	2	2	2	3	3	2	2	2	2	2	2
TN15	3	3	3	3	4	4	3	3	3	3	3	3
TN10	2	2	2	2	3	3	2	2	2	2	2	2
KY3	2	2	2	2	3	3	2	2	2	2	2	2
NC1	2	2	2	2	3	3	2	2	2	2	2	2
TN6	2	2	2	2	3	3	2	2	2	2	2	2
TN7	3	3	3	3	4	4	3	3	3	3	3	3
VA3	2	2	2	2	3	3	2	2	2	2	2	2
NC4	3	3	3	3	4	4	3	3	3	3	3	3
GA2	2	2	2	2	3	3	2	2	2	2	2	2
GA4	2	2	2	2	3	3	2	2	2	2	2	2
TN18	2	2	2	2	3	3	2	2	2	2	2	2
TN20	2	2	2	2	3	3	2	2	2	2	2	2
TN22	2	2	2	2	3	3	2	2	2	2	2	2

VITA

Dana Susan Carney was born in Fort Scott, Kansas, on January 11, 1965. She attended elementary school in the Knoxville City School District and graduated from Bearden High School in June, 1983. She immediately entered the University of Tennessee at Knoxville, and in December, 1986, received the degree of Bachelor of Science in Industrial Engineering. She reentered the University of Tennessee at Knoxville in August, 1989, and in May, 1991, received a Master of Science degree in Industrial Engineering.

She is presently employed as an Industrial Engineering Supervisor with United Parcel Service in Knoxville, Tennessee.