

Southwest Region University Transportation Center

**A Computer Based Methodology for
Optimizing Transit System Station Spacing**

SWUTC/94/721915-1



**Center for Transportation Research
University of Texas at Austin
3208 Red River, Suite 200
Austin, Texas 78705-2650**

1. Report No. SWUTC/94/721915-1		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle A COMPUTER BASED METHODOLOGY FOR OPTIMIZING TRANSIT SYSTEM STATION SPACING				5. Report Date December 1994	
				6. Performing Organization Code	
7. Author(s) Jacqueline Dowds and Randy B. Machemehl				8. Performing Organization Report No.	
9. Performing Organization Name and Address Center for Transportation Research The University of Texas at Austin 3208 Red River, Suite 200 Austin, Texas 78705-2650				10. Work Unit No. (TRAIS)	
				11. Contract or Grant No. DTOS88-G-0006	
12. Sponsoring Agency Name and Address Southwest Region University Transportation Center Texas Transportation Institute The Texas A&M University System College Station, Texas 77843-3135				13. Type of Report and Period Covered	
				14. Sponsoring Agency Code	
15. Supplementary Notes Supported by a grant from the U.S. Department of Transportation, University Transportation Centers Program.					
16. Abstract A technique for determining optimal public transportation system terminal spacing is presented. Following a review of approaches used by other investigators, overall user trip time is selected as a rational viewpoint. User trip times are described mathematically as having access time and line-haul time components each of which are characterized for a range of access and line-haul modes. Relationships and default parameter values are presented in computer software consisting of macro sheets for a commercially available spreadsheet program.					
17. Key Words Terminal Spacing, Intersection Spacing, Population Distribution, Passenger Travel Time, Walk Distance, Wait Time, Acceleration and Deceleration Rates, Bus Schedule Reliability, Average Trip Length, Dwell Time			18. Distribution Statement No Restrictions. This document is available to the public through NTIS: National Technical Information Service 5285 Port Royal Road Springfield, Virginia 22161		
19. Security Classif.(of this report) Unclassified		20. Security Classif.(of this page) Unclassified		21. No. of Pages 181	22. Price

**A COMPUTER BASED METHODOLOGY FOR
OPTIMIZING TRANSIT SYSTEM STATION SPACING**

by

Jacqueline Dowds
Randy Machemehl

Research Report SWUTC 94/721915-1

Southwest Region University Transportation Center
Center for Transportation Research
The University of Texas
Austin, Texas 78712

December 1994

Disclaimer

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated under the sponsorship of the Department of Transportation, University Transportation Centers Program, in the interest of information exchange. The U.S. Government assumes no liability for the contents or use thereof.

EXECUTIVE SUMMARY

Design and operation of public transportation routes and networks inevitably involves decisions about terminal locations. The terminal location question also involves an implicit question about terminal spacing or frequency. Terminal spacing has significant effects upon transit vehicle overall travel speeds and user access times.

Transit terminal spacing has been examined from many different viewpoints including operator costs and user costs or travel times. A review of approaches used by investigators examining transit terminal spacing is presented. A synthesis of reviewed approaches concludes that overall user trip time is a very relevant viewpoint and this approach is used through following elements.

Formulation of a terminal spacing algorithm begins by describing transit user trip time as being composed of access time and on-board time elements. Each of these are described using basic mathematical relationships which are modified to describe wide access mode and line-haul mode ranges. Access modes include walk, auto, and public transportation, while line-haul modes include local bus, bus rapid, light rail, and rapid rail.

Most access and line-haul mode combinations produce user access times that increase with greater terminal spacing but produce line-haul times that increase as terminal spacing increases. For most combinations, one spacing produces a minimum sum of access and line-haul times.

Relationships and default parameter values are housed within computer software which greatly simplifies application. The software is presented in the form of Macro sheets for a very popular generic spreadsheet package. The software communicates with users through dialogue boxes and presents both graphical and tabular results.

ACKNOWLEDGEMENTS

This publication was developed as part of a project study funded by the U.S. Department of Transportation, University Transportation Centers Program supporting the Southwest Region University Transportation Center.

ABSTRACT

A technique for determining optimal public transportation system terminal spacing is presented. Following a review of approaches used by other investigators, overall user trip time is selected as a rational viewpoint. User trip times are described mathematically as having access time and line-haul time components each of which are characterized for a range of access and line-haul modes. Relationships and default parameter values are presented in computer software consisting of macro sheets for a commercially available spreadsheet program.

TABLE OF CONTENTS

CHAPTER 1 Introduction	1
CHAPTER 2 Literature Review	3
CHAPTER 3 The Concept of Minimal User Cost	11
3.1 Minimal User Cost.....	12
3.2 Minimal Operator Cost	13
3.3 Minimal System Cost	14
3.4 User Travel Time	15
3.4.1 Linehaul Modes	16
3.4.2 User Access To Linehaul Modes	17
3.4.3 Vehicle Cycle.....	19
3.5 Summary.....	21
CHAPTER 4 The Mathematical Algorithm to Minimize User Costs	23
4.1 Access Time Parameters.....	23
4.1.1 Catchment Areas	24
4.1.2 Operating Policies	25
4.1.3 Ingress and Egress Time Calculations.....	26
4.1.4 Wait Time Parameter	40
4.2 Linehaul Time Parameters	41
4.3 Summary.....	48
CHAPTER 5 The Computerized Mathematical Algorithm	49
5.1 Program Components.....	49
5.1.1 Program Subroutines	50
5.1.2 Algorithm Inputs	54
5.1.3 Informative Messages.....	59
5.2 Program Sequence.....	60
5.3 Algorithm Results	66
5.4 Summary.....	69

CHAPTER 6 Conclusion	71
6.1 Concepts	71
6.2 Applications	72
6.3 Recommendations	72
 REFERENCES	 73
 APPENDIX A	 77

LIST OF FIGURES

Figure 3- 1	User Access Cost Per Terminal Spacing	12
Figure 3- 2	User Linehaul Cost Per Terminal Spacing	13
Figure 3- 3	Operator Cost Per Terminal Spacing	14
Figure 3- 4	Minimal Total System Cost Per Terminal Spacing	15
Figure 3- 5	Total User Cost Per Terminal Spacing	16
Figure 3- 6	Trajectory Cycle Time	21
Figure 4- 1	Typical Catchment Area.....	24
Figure 4- 2	Walk Catchment Area-Continuous System.....	26
Figure 4- 3	Weighted Average Walk Distance.....	27
Figure 4- 4	Walk Catchment Area-Discrete System	28
Figure 4- 5	Auto Catchment Area.....	30
Figure 4- 6	Local Bus Egress Catchment Area Continuous System.....	31
Figure 4- 7	Local Bus Egress Catchment Area Discrete System.....	33
Figure 4- 8	Local Bus Ingress Catchment Area.....	34
Figure 4- 9	Light Rail Ingress Catchment Area	36
Figure 4-10	Light Rail Egress Catchment Area	38
Figure 4-11	Desired Cruise Velocity Is Not Reached.....	44
Figure 4-12	Desired Cruise Velocity Is Reached	44
Figure 5- 1	Summary Sheet	51
Figure 5- 2	Results File Name Input Box	55
Figure 5- 3	Dialog Box Definition Table	56
Figure 5- 4	First Operating Policy Dialog Box	57
Figure 5- 5	Alert Message Box.....	59
Figure 5- 6	Program Control Flowchart	61
Figure 5- 7	Egress Mode Dialog Box In Typetwo Subroutine.....	63
Figure 5- 8	User Travel Time Costs For An Average Trip Length	67
Figure 5- 9	User Travel Time Costs For The Route Length.....	68

LIST OF TABLES

Table 3- 1	Total System Costs.....	11
Table 3- 2	User Travel Time Components	16
Table 3- 3	Linehaul Modes and Operating Characteristics	17
Table 3- 4	User Access Time.....	18
Table 3- 5	User Access Modes.....	20
Table 3- 6	Vehicle Trajectory Cycle Components.....	20
Table 4- 1	Terminal Spacing Design Procedure	23
Table 4- 2	Ingress and Egress Time Calculation	24
Table 4- 3	Operating Policies	25
Table 4- 4	Weighted Travel Time Per Station Spacing	39
Table 4- 5	Access Modal Split Percents.....	40
Table 4- 6	Wait Time Calculation Options	41
Table 4- 7	In-Vehicle Time Calculation	41
Table 5- 1	Program Subroutines	50
Table 5- 2	Backbone Tasks.....	50
Table 5- 3	Summcalc Tasks	51
Table 5- 4	Typeone Tasks.....	52
Table 5- 5	Typetwo Tasks.....	52
Table 5- 6	Typethree Tasks.....	53
Table 5- 7	Linehaul Tasks	54
Table 5- 8	Input Functions	54
Table 5- 9	Macro Input Formulas	55
Table 5-10	Update Messages.....	59
Table 5-11	While-Next Loop For Segmentation	64
Table 5-12	While-Next Loop For Spacing	65

CHAPTER 1

INTRODUCTION

Public transportation systems provide mobility or an alternative to driving an automobile. Because they enable the public to gain system access, terminals and their locations along transit routes is one of the most important transportation system aspects. Terminals and the distances between them can significantly impact system performance. Consequently, terminals should be located in a way that will provide the most efficient and effective transportation service possible. For the system to be efficient and, thus, desirable, service must be reliable and system access convenient. User needs, therefore, must be kept in mind when designing terminal locations. In addition, the service is effective when user requirements are being met and the system sustains ridership. Once again, user needs are a primary terminal location consideration.

At present, there is no universally accepted method for determining optimal transit terminal spacing. Existing systems have been designed using local policies or rules of thumb and without regard for the relationship between operator and user objectives or for how this relationship should influence terminal location and spacing.

The purpose of this research is to develop a methodology for optimizing the distance between terminals along transit routes that incorporates user needs and requirements. It will set forth a technique to assist with the design of an efficient and effective public transportation system that helps both users and operators achieve their objectives. A computer program will suggest terminal spacings based on this methodology and provide graphical results for use in transit system design or modification.

A transit terminal is any facility where passengers board, deboard or transfer between vehicles or modes. Hence, terminal facilities range from a sign along the street to park-and-ride lots, to concrete structures above or below ground. The simplest accommodate a single mode while the most complex serve as a multi-modal interface. The term "station" is hereafter used interchangeably with "terminal".

The following is a brief description of the remaining chapters. Chapter Two presents a literature review of previous work conducted in this and similar fields. Chapter Three introduces costs associated with a public transportation system. A discussion of the elements of these costs concludes with an explanation of the relationship between them and transit terminal spacing design. Chapter Four translates these concepts into an algorithm for optimizing interstation terminal spacing. In the next chapter, Chapter Five, a comprehensive description of the computer code is presented. The sixth and final chapter summarizes this work, provides implementation recommendations and default value refinement suggestions.

[This page replaces an intentionally blank page in the original document. --CTR Library digitization project]

CHAPTER 2

LITERATURE REVIEW

Several studies have been conducted that focus on transit terminal spacing. Common themes throughout the works include minimizing system costs and reducing required travel times. Following is a brief review of some of those that pertain to this research.

Ghoneim and Wirasinghe (Ref. 6) present a model for use when determining bus stop placement in locations that might not be obvious choices. They define obvious stops to be points of high demand concentration such as shopping malls, central business districts, and intersections with other transit routes. Each is a common destination point shared by several riders. Their model recommends spacing for routes in which there is a "many-to-many" demand. These routes, in other words, cater to riders that come from a variety of origins and have a corresponding number of dissimilar destinations. The model determines optimal spacing to be the inverse of the population density along the route if the objective is minimum access time. To minimize the in-vehicle travel time, the spacing of stops is inversely proportional to the square root of the transit service demand. The authors conclude that ideal stop spacing increases with respect to the number of people on the bus, faster travel speeds, and riding time cost. Spacing decreases, on the other hand, with respect to the cost of walking a unit distance and an increasing demand for boarding and alighting.

Lesley (Ref. 13) analyzed the variables that enter into the location of bus stops for optimal interstop spacing. The passenger variables are in-vehicle time, excess time (walking and waiting), and out-of-pocket costs (transit fares). He states that it is acceptable to use average bus acceleration and deceleration rates and cruising speeds in models since all three vary significantly due to traffic lights and extraneous factors. He assumes that the area which encompasses the trip origins is circular around the bus stop. He equates travel time cost to the average cost per journey or the total journeys per year. Lesley deduces that minimization of these costs by optimizing bus stop spacing is not affected by the value of passenger time or the fare paid.

A model of optimal bus stop spacing for commuter routes is presented by Kuah and Perl (Ref. 12). The model provides a means to design a feeder bus network for a rail station. The main objective of the model is minimization of both user and operator costs. User cost variables are based upon time values for walking, waiting at the bus stop, and time lost on the bus while it is dwelling at subsequent stops. The model assumes a constant transit demand between two adjacent routes because the route spacings are small compared to the service area size. They assume the average distance to the nearest route to be one-quarter of the route spacing and the distance to the nearest stop to be one-quarter of the spacing between stops. The model calculates passenger wait time at the stop as one-half the bus headway. On the other hand, operator costs include running time cost and costs associated with time lost at bus stops.

Kuah and Perl examined three possible cases of bus stop spacing to determine the optimal spacing. The solution to the first two -- uniform stop spacing according to the demand and constant spacing along the route -- varies directly with the average trip length, passenger walking speed, value of passenger riding time, and average time lost per bus stop. The value of each variable increases with greater interstop distances, which are inversely related to walking time value. The third case, variable stop spacing, indicates that optimal stop spacing varies directly with the square root of walking speed and is inversely related to the square root of walking time. The spacing in this scenario increases with increasing values of passenger riding time and average time lost at stops. These results imply that the optimal stop spacing increases as the distance from the rail line decreases. The authors note in conclusion that if stop spacing is constant, it will be insensitive to the demand density along the route.

Wirasinghe presents yet another rail/feeder bus system model (Ref. 30). According to him, it is important to locate feeder bus routes so as to minimize travel time and operating costs. A set of decision rules assigns feeder bus routes to the appropriate rail station. He presents analytical and graphical techniques to design a network for two different scenarios: pre-determined rail station locations and long (relative to the area served) feeder bus routes. The output from these models determines the most logical feeder bus routes and their service frequencies as well as preliminary rail station locations. A test using the Calgary, Canada LRT system proved that a simple analytical model of a complex transit system can provide reasonable results and assist with the design.

In his thesis, Feder (Ref. 5) recognizes the importance of travel time to the financial condition of a bus transit operation and the corresponding need to determine optimal interstop distance. A minimal round trip time on a route is necessary so route demand may be met with fewer buses. At the same time, users desire a travel time that is available at the least expense possible. A transit company has a variety of options available to reduce travel time. These include a reduction in the number of stops, stopped time, and increased vehicle acceleration rates. Ideally, the results of these actions would yield a decreased route travel time as well as increased patronage. However, the reductions mentioned would automatically increase user access and egress times. Feder presents an analytical and graphical procedure to balance this discrepancy and determine the appropriate stop spacing. The data used for calibration purposes was obtained from buses in Pittsburgh, Pennsylvania. The results indicate that placing stops every six to eight blocks minimizes bus travel time as a function of necessary fleet size. Conversely, stops placed every two to four blocks minimize travel time relative to passenger access cost. The author ultimately recommends a five block compromise spacing.

Vuchic (Ref. 25) addresses the optimal number of terminals and their locations for a rapid or commuter rail linehaul passenger transportation system. In this "many-to-one" type of demand, commuters share one common destination point: the central business district. Station locations should minimize total commuter travel time. Vuchic develops analytical and graphical algorithms to model the situation and determine required spacing. He assumes that passengers choose the boarding or alighting

station that will minimize travel time to and from the station. The model incorporates population density along the rail line, access speed, dynamic train characteristics, station dwell time, and intermodal transfer time as parameters. Four scenarios are considered: 1) any population distribution along the line and no restrictions on the interstation spacing; 2) uniform population distribution along the line and no restrictions on interstation spacing; 3) any population distribution along the line with interstation spacings that are greater than the critical distance (the length of line needed for the transit vehicle to reach designed cruising velocity); 4) uniform population distribution along the line with interstation spacings that are greater than the critical distance. The algorithm requires four steps. The first determines the passenger shed. The second computes the access and on-train (in-vehicle) times. In the third step, the set of simultaneous equations that will determine optimal station locations is derived via differential equation or dynamic programming techniques. In the final step, the number of stations and the corresponding interstation distances are obtained. Optimal spacings depend upon the ratio of the number of people waiting to board the train to the number already on the train. The required density of stations, therefore, increases proportionately with this ratio. The simplest case (constant spacing) occurs when the passenger demand is uniform along the entire length of the line and the ratio remains constant. If the population density remains constant, the ratio decreases along the line toward the common destination point (central business district) and the train has to stop less often as it approaches this point. Thus, interstation distances increase toward the destination point. This pattern of cumulative boarding is quite common for radial commuter lines. Various parameters were changed to test the model sensitivity. The station density varies inversely with and is very sensitive to the ratio of passenger access speed to maximum vehicle running speed. It is not very sensitive to the dynamic characteristics of the transit vehicle, however. Ultimately, the model remains most sensitive to the ratio of people waiting to board to people already on the train.

In another study, Vuchic (Ref. 26) analyzes optimal rapid transit interstation spacing with regard to maximum number of passengers. In a model similar to the previous one, he compares two systems. One of the systems is discrete and can be accessed only at transit terminals, or stations, while the other, a continuous system, can be accessed at any point. The model assumes that passengers will select the system that provides the shortest overall travel time. Passengers are uniformly distributed along the line and share the central business district as their common destination point. A similar algorithm determines the ideal station spacing for this objective. The optimal solution is more sensitive to the speed of the continuous system than to that of the discrete. The results also indicate that the area captured by the discrete system grows with faster access speeds. To maximize the number of people using the system, the interstation spacing should increase as the distance to the common destination point decreases. Station density required for this particular objective is greater than that required for minimum travel time. When the algorithm constrains the practical number of stations allowed, however, the solution with regard to passenger maximization approaches that of travel time minimization.

Another approach for deriving the optimal interstop spacing on transit routes is to first determine the ideal number of stops and then divide the route length by this number. The analysis of the optimum number of stops and the optimum vehicle stopping policy for operation of a transit route performed by Vuchic (Ref. 27) considers two primary objectives: minimum user travel time and minimum total cost. It considers three stopping policies: all-stop stopping, on-call stopping, and demand stopping. Vuchic determines that the number of stops made follows a binomial distribution. The average user travel time that is to be minimized is a function of several factors, namely passenger volume, headway, access speed, dynamic vehicle characteristics, and number of stops. For a small passenger volume, the number of stops is theoretically equal to twice the number of passengers since the average user travel time is at a minimum when the number of stops is at a maximum according to Vuchic's model. Realistically, a demand-stop operation best suits this condition. A large passenger volume is the other extreme condition that was analyzed. The optimal number of stops in this scenario is independent of the passenger volume or the headway and suggests that it approaches a constant that is a function of the route length, average user travel distance, access speed, and time lost at stops. With regard to headway, the number of stops increases as the headway becomes shorter as long as the passenger volume is light to moderate. Otherwise, headway has little influence on the appropriate number of stops. As with passenger volume, the number of stops becomes constant when the transit vehicles are at or near capacity. Underutilized capacity does not influence the number of stops. Moreover, faster access speeds reduce the optimum number of stops. Because a smaller number of stops increases the overall operating speed, the number of stops also influences the required fleet size and related capital and operating costs. Hence, measures to reduce travel time also reduce minimum total costs for large passenger-volume conditions. The optimum stopping regime varies from demand-stop stopping to on-call stopping to all-stop stopping as passenger demand increases. In conclusion, Vuchic recommends that these results be applied when improving existing transit systems or planning new systems.

"Evaluation of A Public Transportation Level of Service Concept" by Polus and Shefer (Ref. 19) shows that route length, stop spacing, number of signalized intersections, and other variables can predict the transit system level of service. A high level of service requires frequent, reliable, accessible, and convenient service. The authors obtained data from twenty-one bus lines in Haifa, Israel in order to develop a way to express this level of service in terms of travel time measures. Analysis indicates that the level of service increases with an increase in the average distance between stops. It decreases, on the other hand, with an increasing number of signalized intersections. The data was used to calibrate level of service prediction models that are transferrable to public transportation systems anywhere.

The Urban Bus Transit: A Planning Guide (Ref. 23) manual suggests that the most important of these variables are user-related service attributes. The first of these is travel time, which consists of the time a passenger spends walking to and from the bus stop, waiting for the bus, riding on the bus, and transferring between buses. The convenience attribute accounts for the frequency of bus service, the

service hours, and the number of transfers required to reach an ultimate destination. The presence of shelters at stops, characteristics of the transit vehicle itself, and the degree to which vehicles are crowded are all indicators of the comfort level. Furthermore, the delays that occur on the route and the schedule adherence capabilities that are perceived by users help to form their opinion of the system reliability. The fare required to use the system is the final user-related service attribute to be considered when developing a transit system that will provide an acceptable level of user service.

Non-monetary user costs stem from access time to and from the transit stop, waiting time at the stop, and in-vehicle travel time. According to Seneviratne (Ref. 21), the typical access mode to bus stops is walking and there is a maximum distance that passengers are willing to walk to and from a stop. It is within these distances that transit stops, parking facilities, etc. should be located. A survey of 2685 randomly chosen people in downtown Calgary, Canada was conducted to determine the travel paths they had taken to reach the area and the reasoning behind their choices. Trip purpose is a crucial element for determining the needs of the pedestrian and, hence, the acceptable walking distance. A person on a shopping or leisure trip is willing to walk farther than a commuter going to work who, in turn, is willing to walk farther than a person on a business or errand trip. A cumulative frequency distribution plot of the 2685 walking distances determined the maximum acceptable walking distance to be the maximum rate of change of the slope. At this point, the smallest change in walking distance will affect the greatest number of people. This critical distance understandably varies if the majority of pedestrians are shoppers as opposed to commuters. The author concludes that this method is useful even when information is limited; it is also useful because it takes into account the needs and feelings of people that influence their tendency to walk and take transit or choose another travel mode.

A similar study (Ref. 21), also conducted in Calgary, suggests that existing transit stops are typically distant from shopping centers and industrial areas. This obviously favors automobile use over transit use since the locations require passengers to exceed their preferred walking distances. The findings of this study further indicate that the transit stop catchment areas should be redefined since most of the current stops exclude substantial portions of a majority of Calgary's residential neighborhoods. Ideally, the boundaries should fall within the maximum acceptable walking distance of potential transit users.

Petersen (Ref. 18) determined these acceptable walking distances for Washington D.C. residents based on car ownership and socioeconomic status. Data obtained from bus riders commuting to work in the area indicates that there is a distinct difference between automobile owners and non-automobile owners. The range of acceptable walking distances based on socioeconomic levels is much larger for those who do not own automobiles than for those who do. The owners accept a maximum walking distance that is around the midpoint of the previously mentioned range. The results also prove that the tendency to use transit decreases as the distance from the stop increases.

Access cost is based on access time. Feder (Ref. 5) analyzes the access time and its conflict with optimal stop spacing. An optimal walk distance for each passenger would result in a large number of closely spaced stops. However, a large number of stops is not cost effective for the transit operator, so Feder recommends a compromise spacing of two to four blocks to minimize access time and cost. When access times for different spacings are computed, the results indicate that the cost of travel time increases less significantly over the previous spacing once a spacing of five blocks is reached. Feder approximates the walk distance to be within the region that encompasses one-half the distance between stops parallel to the transit route and one-half the distance between routes perpendicular to the transit route.

Another component of user cost is the time spent at the stop waiting for the transit vehicle. Tozzi and Abkowitz (Ref. 2) address the issue of headway-based reliability control. Headway control involves holding buses at certain points until the minimum headway time is reached. This provides more reliable service to the transit users since buses cannot arrive early if controlled in this manner and cause passengers to miss the bus. Enhanced schedule reliability would permit passengers to plan accordingly so as to minimize their time and cost associated with waiting at the bus stop. The findings imply that routes in which the majority of passengers board and alight at the middle or near the end of the route benefit most from headway control and produce the most significant reduction in passenger wait times, or costs.

Levinson (Ref. 14) investigated factors that influence in-vehicle journey time and makes suggestions for improving transit performance. Studies of bus transit systems in Boston, Chicago, New Haven, and San Francisco indicate that car speeds are consistently 1.4 to 1.6 times faster than bus speeds in similar conditions. In addition, buses spend nine to twenty-six percent of the travel time at passenger stops and twelve to twenty-six percent in traffic delays. This large percentage of time spent stationary at stops should be reduced in order to be more competitive with the automobile mode. Further, the buses averaged a travel time of 4.2 minutes per mile (min/mi) in the suburbs, 6.0 minutes per mile (min/mi) in the city, and 11.5 minutes per mile (min/mi) in the central business district. Levinson derives the travel times and speeds as a function of the stop frequency, stop duration, and bus acceleration and deceleration times. These are all dependent on the number of stops on the route and the dwell times at each stop. He shows that the number of stops actually made decreases with decreasing population density away from the central business district and helps to account for reduced dwell time. Recommendations to improve transit performance include minimization of the number of stops made, wider door widths on the transit vehicle, and a more efficient fare collection method.

A study conducted in Halifax, Nova Scotia by Seneviratne and Loo (Ref. 22) indicates that the effects of stop spacing and dwell times on bus journey times are a function of the area type (central business district, inner-city, or suburbs) and the corresponding land use patterns. Although variability in journey time is partially determined by time spent waiting for passengers to board, a sudden demand surge at one or two particular stops is unnoticeable because of slack time built into schedules. The authors conclude that the major components of overall journey time are stop delays (including the time to

accelerate and decelerate) and actual dwell time while extraneous factors such as traffic signals, congestion, accidents, and construction delays are only a small proportion of the travel time variation.

A more thorough knowledge of dwell time and its route performance impact can yield suggestions to decrease it as well as produce more accurate service schedules and terminal berth requirements. Kraft and Bergen (Ref. 11) use data collected from Louisville, Kentucky along with the method of least squares to develop equations to predict passenger service time for three scenarios: passengers boarding, passengers alighting, and simultaneous passenger boarding and alighting. They develop separate equations for the cash-and-change fare system and the exact fare collection method. They conclude that midday passenger service times are greater than for either peak period, that local transit service requires less loading and unloading time than intercity service, and that the exact fare method saves 1.4 to 2.6 seconds of dwell time per passenger.

An examination of bus delay caused by stopping and starting at bus stops and by passenger boarding and alighting was conducted by Guenther and Sinha (Ref. 7). Their findings, based on a study of two urban bus routes in Milwaukee, Wisconsin, indicate that the negative binomial distribution makes accurate predictions of numbers of people that board and alight at stops along the route. The study suggests that dwell time per passenger is dependent upon the number of passengers boarding and alighting at the stop in that the time decreases with the natural logarithm of the number of boardings and alightings. Regression analysis performed on the data suggests that dwell time also depends upon other factors such as fare structure, vehicle size and number of doors, and passenger physical capabilities. Per stop dwell time was at a maximum when there were approximately twenty-four passengers boarding or alighting. Guenther and Sinha note that time per passenger decreases as the number of passengers boarding and alighting increases even though the total dwell time increases. From the passenger distribution and dwell time analysis, they estimate bus delay time as a function of these two parameters and use it to compute bus operating speed. A comparison showed that the model results were within ten percent of the actual values recorded along the two routes. The authors conclude that a change in the number of posted stops along a low demand route will have a minor effect on the bus operating speed and will decrease the user access distance. Such changes have even less effect on high demand routes.

Urban Bus Transit: A Planning Guide (Ref. 23) is a comprehensive presentation of all aspects of the design of an urban transit system. It discusses the need for a quality transportation system and presents in great detail all aspects from designing and initiating a new system to the inherent public policies. The basic elements of an urban transit operating system are demand, supply, and operator objectives. "Demand" is the set of ridership estimates which depend on fare amount, headways, and route location. It interacts with the operating system to yield the service characteristics (frequency, travel time, area served, hours of service, etc.) of the system. The transit company, buses, drivers, and mechanics create the transit "supply". The supply and demand components interact together with the operating system to produce cost measures that account for the area's resources that are consumed by

the system. Finally, "operating objectives" are the policies that define the acceptable trade-off between the cost of these resources and the number of rides provided. The three stages of planning used to balance the demand and supply in order to meet the objectives are strategic, functional, and route design.

According to the guide, the cost of the transit properties is another important aspect of the urban transit system to be considered. "Driver costs" cover employee wages and benefits as well as premiums and other allowances. "Vehicle costs" are related directly to the number of vehicles required for service while "transportation costs" are the vehicle costs related to the actual number of miles driven. These costs depend upon the age of the fleet, type of equipment used, and the active maintenance and revenue standards. Furthermore, there are fixed overhead costs associated with the administration and support of the transit system and capital costs that account for the annual amortization costs of facilities and equipment.

Attention given to the subject of optimizing transit system performance and the consequential levels of service they provide to the public indicates how important these systems are to urban area welfare. This literature review is intended to reflect this importance and thus includes a variety of possible approaches that concentrate on optimal stop spacing for both bus and rail modes in order to meet user needs. In addition, the review included studies about user behavioral characteristics.

CHAPTER 3

THE CONCEPT OF MINIMAL USER COST

The total cost of a public transportation system can be described as the sum of costs associated with system operations, and includes user costs. An understanding of this cost is necessary in order to produce an effective design that encompasses user-defined criteria. Hence, the sum of user and operator costs comprise the total system cost (Table 3-1). The former are those encountered when using the transportation service. Users usually pay a fare each time they use the service or purchase time or trip based passes. Another cost incurred by riders is time required to complete the journey via the system. Finally, any loss of mobility or inconvenience the user experiences because of using public transportation adds to the expenditure. Levinson (Ref. 14, 1), for instance, conducted a study in several large

Table 3-1 Total System Costs

Total System Cost =

User Costs	+	Operator Costs
Monetary Fare		Fleet Size/Transportation
Travel Time		Overhead/Capital
Inconvenience		Employee Wages

metropolitan areas and concluded that automobile speeds are consistently 1.4 to 1.6 times faster than bus speeds in similar traffic conditions. The patron pays for this inconvenience when using the public transportation service.

On the other hand, several elements comprise the expense of providing a transportation service (operator cost). Foremost are vehicle and transportation costs. The fleet size required and the facilities to house it dictate the vehicle costs while those related to the actual miles driven create the transportation costs. These include fuel, maintenance, vehicle repair, and facilities to support these operations. Moreover, there are overhead costs that result from administration and support services and their facilities, advertising, schedule publishing, and terminal maintenance. Capital costs account for the annual amortization of physical facilities. Employee salaries and benefits add the final element to the total expenditure necessary to operate a transit system (Ref. 23, 144-145). These costs must be included with those of the user when computing total system cost.

The most economical design produces a transportation system that attains the desired levels of efficiency (from the user standpoint) and effectiveness (from the operator viewpoint) for the lowest possible system cost. Since transit terminal spacing has a significant impact on these two measures of system performance (to be explained subsequently), terminal spacing should reflect an intent to minimize costs without sacrificing performance.

3.1 Minimal User Cost

Minimal user access cost can be attained through minimal terminal spacing. Time spent ingressing and egressing the system at the terminals and time spent riding in the vehicle are the two user time components. Riders spend less time accessing service when terminals are closer together because they have a shorter distance to travel from trip origin to terminal and from terminal to trip destination. Figure 3-1 portrays this concept. The terminal spacing (feet) is on the horizontal axis and user cost in terms of the trip time minutes not in the transit vehicle is on the vertical axis. The graph clearly illustrates that out-of-vehicle user cost increases as distance between terminals increases. Thus, a higher terminal frequency yields minimal cost.

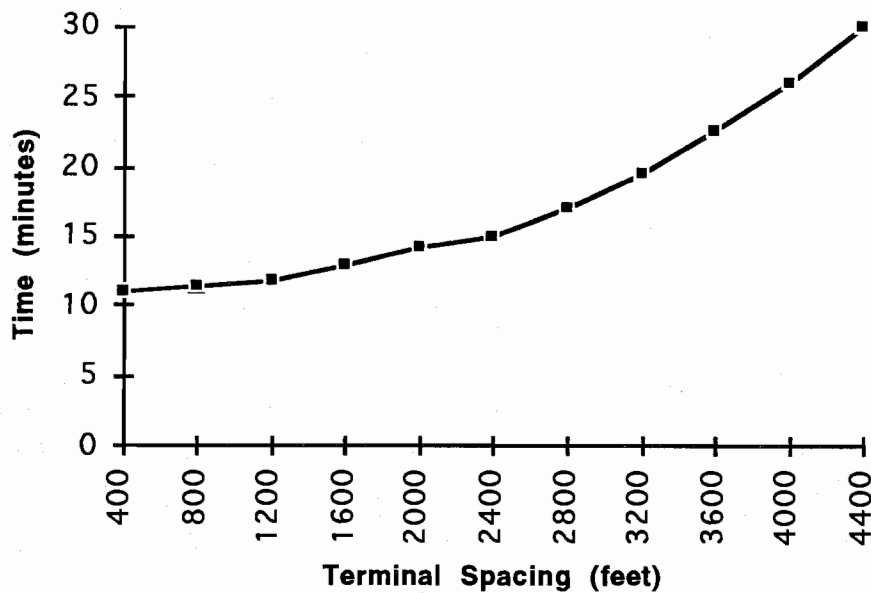


Figure 3-1 User Access Cost Per Terminal Spacing

Conversely, time spent in the vehicle becomes large if the terminals are closely spaced because the vehicle will stop more often and will sustain a lower cruising speed. As Figure 3-2 illustrates, user costs increase with decreasing interstation spacings. Again, the distance between terminals is shown in feet on the horizontal axis while the user cost in minutes spent riding in the vehicle is on the vertical axis. As a result, a system that spaces terminals farther apart minimizes in-vehicle time.

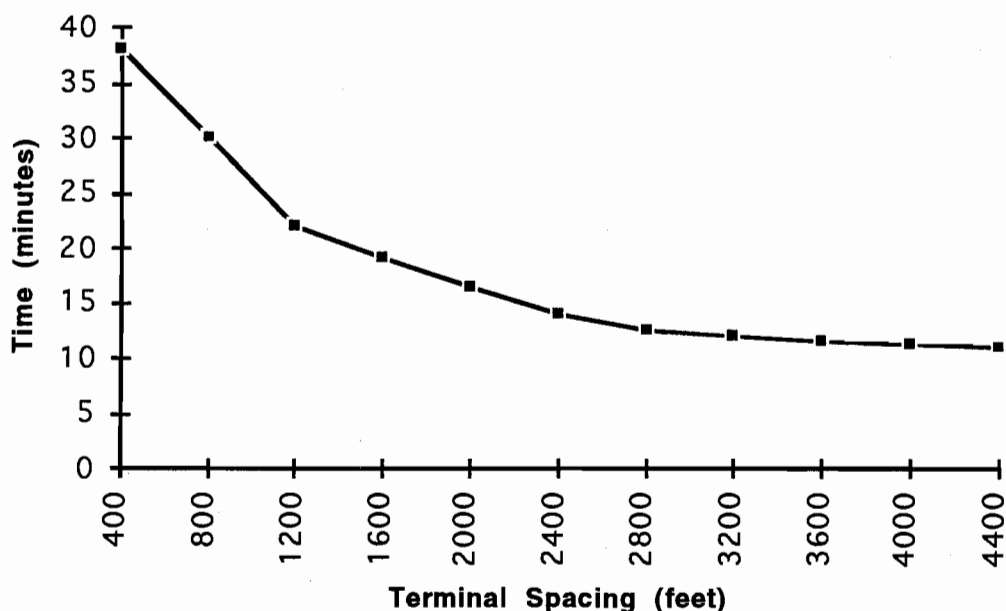


Figure 3-2 User Linehaul Cost Per Terminal Spacing

Although these two approaches to terminal frequency conflict, research presented in Urban Bus Transit: A Planning Guide (Ref. 23, 171) indicates that the public values out-of-vehicle travel time two to three times more heavily than in-vehicle time. In other words, people are willing to incur a cost that is three times as great to save out-of-vehicle time than to save time riding in the vehicle. Hence, users accept the extra time spent stopping at more terminals so that they will have shorter ingress and egress distances to and from terminals. Therefore, the ideal design for minimal user cost favors more terminals spaced closely.

3.2 Minimal Operator Cost

A minimal cost scenario for the operator, on the other hand, places fewer terminals with longer interstation spacings along the transit route. Figure 3-3 depicts this scenario. The interstation distance (feet) is on the horizontal axis while operator time cost (minutes) to complete one pass over the route, is on the vertical axis. With this type of terminal location design, fewer stops by the transit vehicle are

necessary due to fewer terminals. The operator experiences a savings in time and cost since stop delays (the time to accelerate and decelerate) and actual dwell times, which are major overall journey time components (Ref. 22, 272), are minimized.

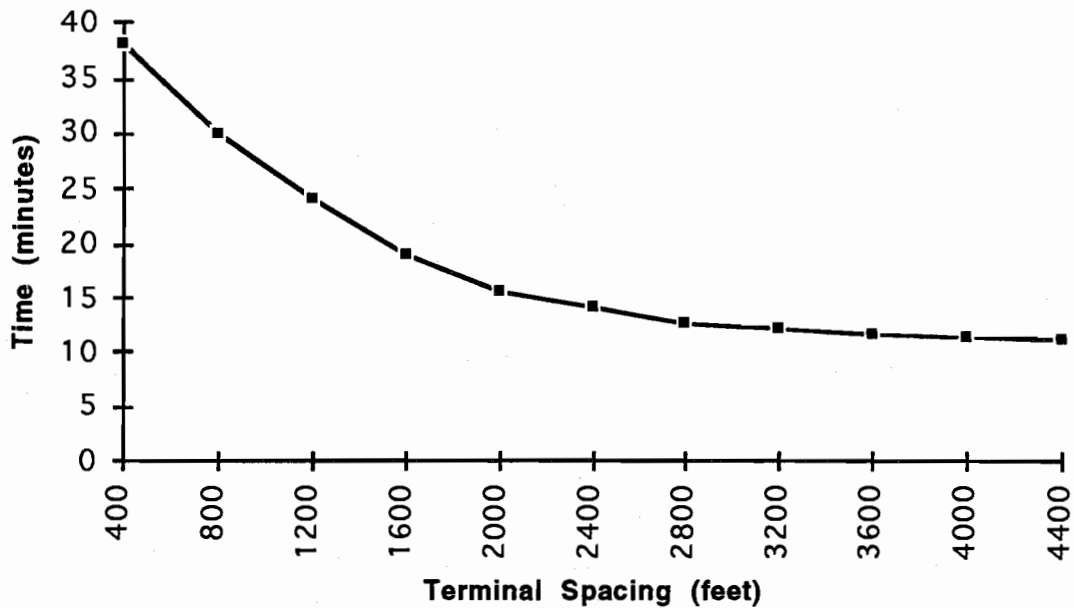


Figure 3-3 Operator Cost Per Terminal Spacing

With fewer required stops, the system operator realizes a time savings because more time is spent at cruising speed and less time is spent accelerating and decelerating. Furthermore, there is less overall dwell time although a slightly longer dwell time at each stop seems logical since more passengers would be waiting than if there were more access points. However, this is not the case. According to Guenther and Sinha (Ref. 7, 9), the boarding time per passenger decreases as the number of passengers boarding increases. Therefore, the overall dwell time decreases along with the time to complete one pass due to the reduction in the number of times the vehicle has to deviate from cruising speed, open and close the vehicle doors, and merge back into the traffic stream. All of this results in a time and cost savings for the operator when the interstation distance is great and the terminal frequency small. There are also fewer terminal facilities to construct and maintain in this spacing design. A reduced amount of fuel and wear and tear on the vehicles (brakes, doors, etc.) further adds to the operator savings. Fewer terminals spaced farther apart consequently produce the minimal operator cost.

3.3 Minimal System Cost

Station spacing, therefore, affects the level of service provided by the system and the subsequent attraction to use it (Ref. 19, 140-141). Ideal spacings are very different for users and operators, however, so it is necessary to reach a compromise. Respective cost curves are shown in Figure 3-4 with spacing (feet) on the ordinate and cost again represented by minutes on the abscissa. The third curve represents total system cost and is the sum of the user and operator cost curves. The smallest ordinate on this curve reveals a compromise solution for station spacings that yields minimum possible cost.

While this design may be acceptable and efficient enough for some potential users, it is not necessarily effective since calculation of this operator cost does not take into account any user-oriented criteria. This spacing design, therefore, does not yield the most efficient and effective transportation system. One that does must focus on user cost and account for user needs and preferences so it will initiate and sustain ridership.

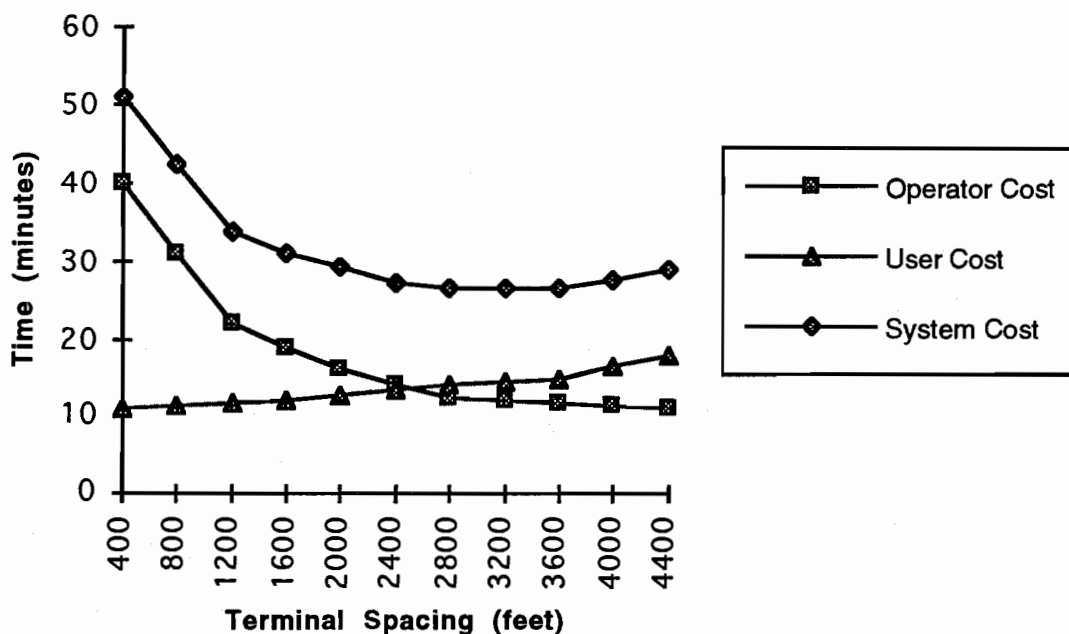


Figure 3-4 Minimal Total System Cost Per Terminal Spacing

3.4 User Travel Time

Since trip time is the most significant portion of user cost, user travel time is the foremost item to consider when designing the transit terminal spacing (Ref. 23, 167). This methodology, therefore, minimizes user cost in terms of travel time to determine the optimal interstation spacing. As shown in

Table 3-2, the two components of the travel time are access (out-of-vehicle) time and linehaul (in-vehicle) time. Figure 3-5 shows the related cost curves and their sum.

Table 3-2 User Travel Time Components

Access (Out-of-Vehicle) Time

Linehaul (In-Vehicle) Time

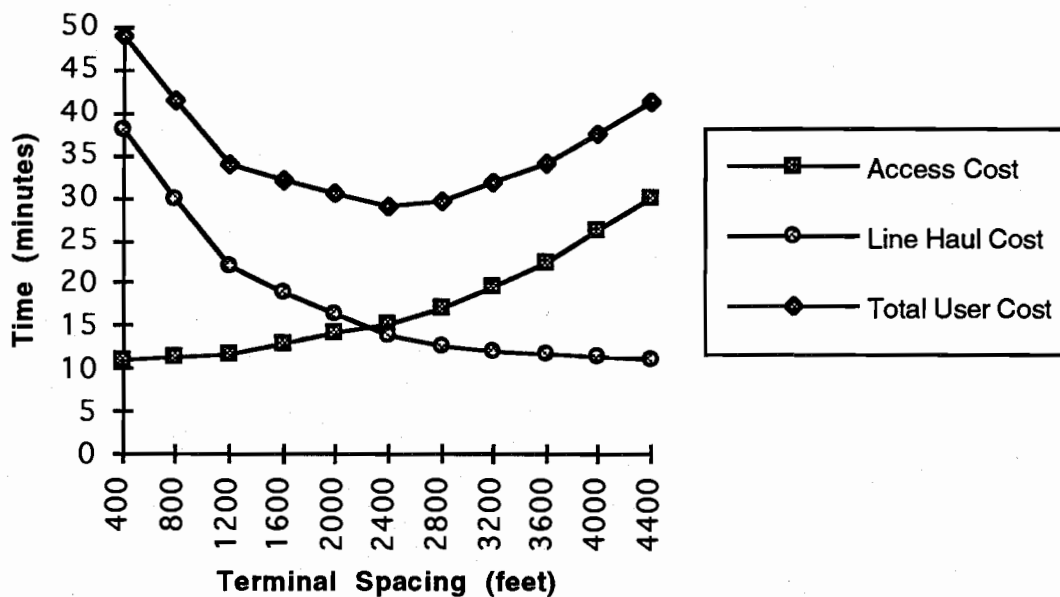


Figure 3-5 Total User Cost Per Terminal Spacing

As in Figure 3-4, the minima of the sum curve represents the compromised spacing for the lowest possible cost based on the user-oriented criteria.

3.4.1 Linehaul Modes

The linehaul time is the time spent on the primary mode used to make the transit trip. This methodology deals with four transit modes: local bus, bus rapid, light rail, and rapid rail. The local bus operates on urban streets in mixed traffic (Ref. 27, 191) and shares right-of-way with other vehicles. Vehicle capacities range from 70 (standard bus) to 125 (articulated bus) with passengers sitting and standing. Local bus ride quality is the lowest of the four modes.

The bus rapid mode provides a higher ride quality. It uses the same type of vehicles as the local bus mode, but has partially reserved right-of-way available. Time savings due to this right-of-way arrangement comes from exclusive use by buses and carpools of a high occupancy vehicle lane (HOV). Vehicles in this lane are not delayed by typical traffic congestion and slow speeds during peak periods. Another form of this mode is express bus service, which stops only at selected terminals usually at the collection and distribution route ends (Ref. 27, 243).

The light rail mode operates on a right-of-way that is anywhere from 40% to 90% separated from other street traffic. Power distribution is via overhead wires and vehicles can accommodate 170 to 250 sitting and standing passengers. Its ride quality is superior to that of local bus.

The rapid rail mode provides the highest ride quality of the four modes. It operates on a completely separated right-of-way and can, therefore, achieve the highest speeds. This separation also allows for power distribution in the rails. Unlike the other modes, trains do not operate on the urban streets or highways. Vehicles can transport anywhere from 120 to 250 sitting and standing passengers. Table 3-3 summarizes the four modes and their operating characteristics.

3.4.2 User Access To Linehaul Modes

As Table 3-4 indicates, access time is the time spent traveling to the linehaul mode terminal from the origin and from the ensuing linehaul terminal to the destination along with time spent at terminals

Table 3-3 Linehaul Modes and Operating Characteristics

Linehaul Mode	Right-of-Way	Capacity	Ride Quality
Local Bus	Non-Exclusive	70 - 125	Lowest
Bus Rapid	Partially Exclusive	70 - 125	Best of bus modes
Light Rail	Partially Exclusive	170-250	Lower than rapid
Rapid Rail	Exclusive	120-750	Highest

waiting for the vehicle or transferring between vehicles or modes. The mode employed to travel to and from the linehaul terminals is the primary ingress mode while the secondary ingress modes are those used to access primary modes. Each linehaul mode has a unique set of primary and secondary access modes. A more detailed explanation of the time required for each follows.

Table 3-4 User Access Time

Ingress Linehaul Terminal

Wait for Linehaul Vehicle

Egress Linehaul Terminal

The primary ingress mode to the local bus is walking (Ref. 18, 28). Signs that indicate the bus route name and number demarcate local bus terminals that may also include a bench and a simple shelter. The time, or cost, is the time for the user to walk from the trip origin to the terminal and then wait for the bus. At the destination end, the egress modes are walking, or riding another local bus if transferring is necessary. The time cost for the latter involves waiting for the next bus, riding on it, and walking from that egress point to the journey destination.

There are several ingress and egress modes for rapid bus. Terminals are typically suburban, park-and-ride lots, simple stops with signs along streets, or major central business district transfer facilities. The following discussion refers to passengers boarding in the morning near their residence and alighting near their work place. Ingress and egress modes reverse for the afternoon commute. A few rapid bus riders walk to ingress points while the majority drive to parking lots (Ref. 17, 23). The ingress time (and corresponding cost), therefore, consists of the time needed to enter the automobile, drive to the terminal facility, and walk to the loading point from the parking space. A collector/distributor (feeder) bus is another common suburban ingress mode (Ref. 12, 341). Time to walk from the origin to the feeder bus terminal, wait for the bus, and ride the bus comprise the cost for this ingress mode. In larger cities with dense populations, light rail might be a bus rapid ingress mode option. The secondary ingress modes would be walking, driving, or riding a feeder bus to the light rail terminal in this case. Passengers must then spend time waiting for the rapid bus. With the exception of walkers, users incur the time and additional cost to transfer and wait for the primary egress mode. These are typically local buses or light rail modes. The secondary mode from these is walking to the final destination.

Ground level platforms or raised concrete islands with a covering for shelter serve as terminals for light rail vehicles. In the suburbs, these are located in areas similar to bus rapid park-and-ride lots and,

hence, the primary and secondary ingress modes are similar to those for bus rapid (Ref. 16, LR4). System patrons walk, drive, or ride a feeder bus to the terminal. Accordingly, ingress costs stem from the time required to travel by these modes, access the automobile or feeder bus, and then wait for the light rail vehicle. Egress modes at the destination end in the central business district can be walking or transferring to a local bus. Egress terminals are usually located at transfer centers providing convenient access to local buses or may be stops along the street. Once again, the ingress and egress modes reverse for the return trip.

Finally, rapid rail stations are generally underground in the central business district or city center and above ground in the suburbs where sufficient land is available. Here again, ingress modes include walking, driving an automobile, riding a collector/distributor bus, and in dense areas, perhaps a feeder light rail line. Walking and transferring to a local bus or light rail route are the modes used to egress to the destination point. The opposite is then the case for the return trip. The access time and costs are identical to those outlined for rapid bus. Table 3-5 summarizes these modes, listing the primary modes with secondary modes following the slash.

3.4.3 Vehicle Cycle

In-vehicle costs plus out-of-vehicle costs comprise total user cost. Time to travel on the linehaul mode accounts for the in-vehicle cost. The number of vehicle trajectory cycles between ingress and egress points determines this time. A cycle, as shown in Table 3-6, consists of four parts (Ref. 27, 157-167). First is the time the vehicle requires to accelerate from zero velocity at the terminal to the cruising speed. Line segment **a-b** on Figure 3-6 illustrates the time the vehicle takes to reach the desired speed. Segment **b-c** represents the time spent cruising at a constant velocity, or the second cycle component. Segment **c-d** indicates the time to decelerate (third cycle component) the vehicle to zero velocity and stop at the terminal. The dwell time (**d-e**) is the stationary time spent loading and unloading passengers at the terminal and is the final cycle component. The portion of the axis between **a** and **d** is the actual travel time between terminals. The length of the time axis between **a** and **e** shows the time to complete the vehicle cycle since **e** represents the time at which the driver initiates acceleration from zero velocity and, thus, begins another cycle. The trajectory cycle time is not always constant over the entire route for the bus and light rail modes since the cruising velocities and dwell times may change on various route segments. The rapid rail cycle time, on the contrary, is constant since these cycle components can be fixed.

Table 3-5 User Access Modes

Linehaul Mode	Ingress Modes	Egress Modes
Local Bus	Walk	Walk Local Bus/Walk
Bus Rapid	Walk Automobile Feeder Bus/Walk Feeder Rail/Walk- Automobile-Bus	Walk Local Bus/Walk Light Rail/Walk
Light Rail	Walk Automobile Feeder Bus/Walk	Walk Local Bus/Walk
Rapid Rail	Walk Automobile Feeder Bus/Walk Feeder Rail/Walk- Automobile-Bus	Walk Local Bus/Walk Light Rail/Walk

Table 3-6 Vehicle Trajectory Cycle Components

<p>Accelerate</p> <p>Cruise</p> <p>Decelerate</p> <p>Dwell</p>
--

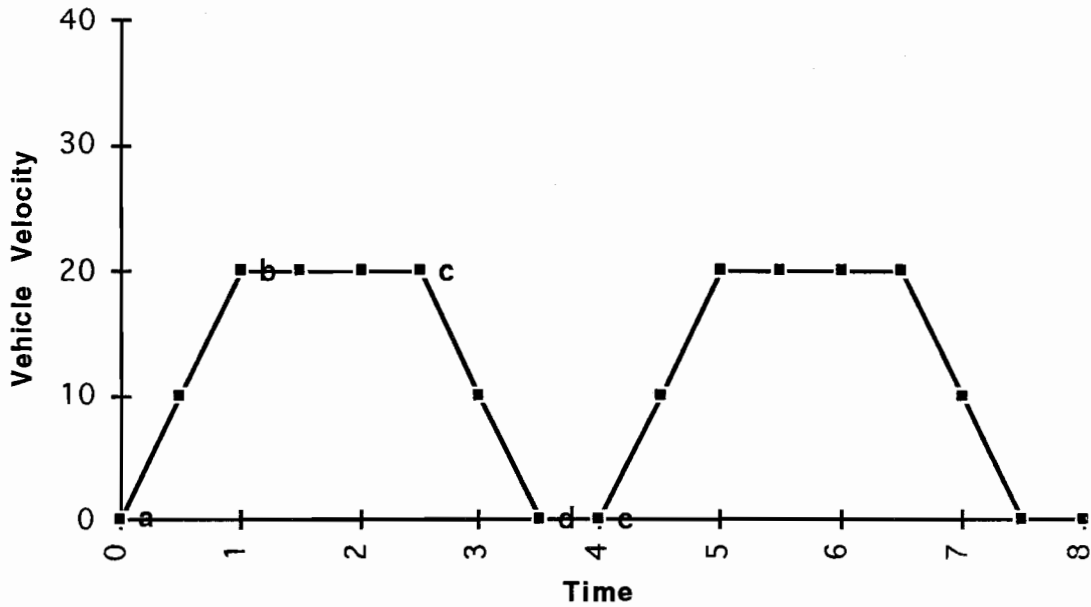


Figure 3-6 Trajectory Cycle Time

Although vehicles must stop for traffic lights and construction detours, these variations to the travel time are insignificant compared to deviations from cruising time for accelerating/decelerating and boarding/alighting passengers. These factors, therefore, are not included in the determination of user in-vehicle time and cost.

3.5 Summary

Knowledge of public transportation system usage procedures is necessary in order to accurately predict journey times. Because terminal spacing and frequency significantly impact this time, it is beneficial to optimize spacing to minimize user cost. The total user time, or cost, equates to access time plus linehaul time. Minimal user times for these two times, respectively yield different terminal spacings. A compromise between the two different spacings must therefore be reached since this time is the most prominent portion of the cost to use the transit system. The compromise is obtained by computing the access and linehaul costs and summing them for a variety of possible station spacings. The shortest of these times provides the least user cost and the optimal interstation spacing. User travel time is the key element to a methodology that optimizes transit terminal spacing by minimizing user cost.

CHAPTER 4
THE MATHEMATICAL ALGORITHM TO MINIMIZE USER COST

Now that the constituents of user travel time have been identified, they will be used to construct a mathematical algorithm that computes total travel time. As Figure 3-5 shows, each station spacing costs the user a unique amount. This algorithm, then, calculates the user travel time per interterminal spacing over an appropriate distance range. The shortest of these times equates to the lowest user cost and the corresponding spacing yields the optimal interstation spacing from the user perspective. With this method, a terminal spacing design for the system results that is both efficient and effective according to the user-oriented criteria. Table 4-1 chronologically outlines the design procedure.

Table 4-1 Terminal Spacing Design Procedure

Compute Individual Time Components
Compute Total Time Per Station Spacing
Determine Shortest Travel Time
Equates To Lowest User Cost
Produces Optimal Spacing

This methodology either calculates each component of the time required to use the system or represents the component with a value obtained from the literature. These are calculated or represented in the same manner across the four linehaul modes. Once the value of each is determined, it is combined with other appropriate parameters to build the algorithm equations.

4.1 Access Time Parameters

Access time is time spent ingressing and egressing the linehaul terminal and waiting for the linehaul vehicle. As Table 4-2 demonstrates, the ingress or egress time is calculated by dividing the ingress or egress distance to or from the terminal by the appropriate speed. The speeds used in this methodology are values found in the literature while the distances are either calculated based on a catchment area or the actual distance.

Table 4-2 Ingress and Egress Time Calculation

$$\text{Time} = \text{Distance} / \text{Rate}$$

4.1.1 Catchment Areas

People that have a choice will not patronize a transit system if they have to travel farther than what they consider to be a reasonable distance to or from the terminal (Ref. 18, 28). This distance defines an area around the terminal that is known as the catchment area (Figure 4-1). The origin and destination points for a significant majority of transit users are within this catchment area. In this methodology, if the travel distance is to be calculated, the distance from the transit route to the outer edge of the catchment area is labeled the *W*-value to represent a maximum acceptable access distance. On the other hand, the *W*-value specifies the actual travel distance if this method is chosen in lieu of calculating an average maximum travel distance for the entire catchment area.

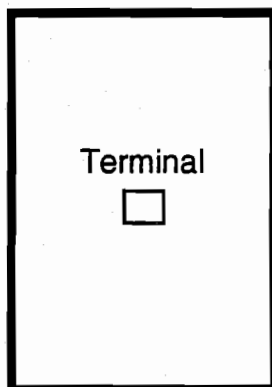


Figure 4-1 Typical Catchment Area

These areas vary for each ingress and egress mode according to the hierarchy and speed of the chosen mode. For example, system patrons will not walk as far as they will drive an automobile or ride a collector/distributor bus because the speeds and distances the latter covers are greater. Furthermore, the length of the entire journey also affects the size of the catchment area for each mode. As the access distance becomes a larger percentage of the overall length of the journey, the user chooses a faster ingress/egress mode, thus allowing for a larger acceptable access distance and allied catchment area.

4.1.2 Operating Policies

The catchment area boundaries for each ingress and egress mode are consistent from linehaul mode to linehaul mode within each operating policy. This algorithm supports three operating policies, which are summarized in Table 4-3. The first is continuous in that the passengers can board and alight anywhere along the route. This scheme applies to local bus routes in cities with smaller transit systems. Next, the second operating policy type is a discrete system and, hence allows boarding and alighting only at established terminals. This is the most common one and applies to all four linehaul modes. The final type, three, is both since it permits boarding only at terminals in the suburbs and city center and deboarding anywhere in the central business district. Some local bus and light rail routes operate under this policy. The catchment areas are much smaller for the continuous systems because users can access the route very close to their origin and destination points.

Table 4-3 Operating Policies

Type	Operation	Modes
One	Continuous (Stop Anywhere Along The Route)	Local Bus
Two	Discrete (Stop Only At Terminals)	Local Bus Bus Rapid Light Rail Rapid Rail
Three	Discrete in Suburbs/ Continuous in CBD	Local Bus Light Rail

4.1.3 Ingress and Egress Time Calculations

Walk As The Primary Access Mode

Walking is a primary ingress and egress mode for every linehaul mode. Of all the linehaul modes, it has the smallest catchment area. People judge a distance to be acceptable for walking if it is a small percentage of the journey length and if the level of service, or ride quality, they receive is worth the effort. This distance also tends to vary greatly with trip purpose and geographical location (central business district, city center, and residential areas). For local bus mode, for example, 1320 feet is a representative maximum acceptable walking distance while people will walk up to 3000 feet to reach a rapid rail terminal (Ref. 24, 18). Therefore, the size of the walk catchment area and, hence, the distances people are willing to walk vary among the linehaul modes.

Continuous System

In this algorithm, the catchment areas for the walk mode are rectangles around transit stations or deboarding points. Figure 4-2 shows the rectangle for a continuous system. The width of the area parallel to the desired deboarding point (marked with an X in the figure) is very narrow due to an operating policy that allows for boarding and alighting anywhere along the route. This bandwidth (BW) is a characteristic of the street system and the physical limitations the surrounding development imposes.

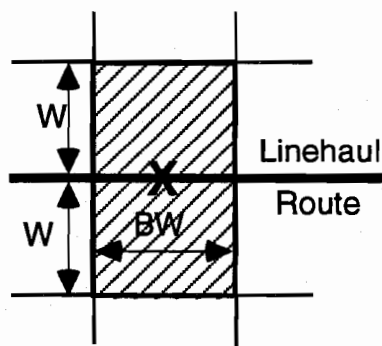


Figure 4-2 Walk Catchment Area-Continuous System

Systems operating under this policy typically provide one stop every block in areas with high trip end densities (central business district). It is not feasible for the vehicle to stop at a different point for every passenger when the volume is as high as it is in these areas. Furthermore, physical constraints like parking lanes and driveways dictate where the vehicle can stop. As such, this bandwidth (BW) parameter recognizes that transit vehicles will not always stop exactly where the rider wants to deboard and, therefore, represents the distance between route signs or the walking distance component parallel to the

route path. Half this distance is chosen as representing all passengers. The W -value, or distance people are willing to walk to access the system, is measured perpendicular to the route on both sides.

The following formula determines the walk distance based on the W -value and bandwidth (BW) parameters. The units of both values and, hence, the result are in feet.

$$\text{walk distance} = \sqrt{W^2 + \left(\frac{BW}{2}\right)^2}$$

distance in feet

The walking distance is the length of the hypotenuse of a triangle (Figure 4-3) with the W -value and a half bandwidth $\left(\frac{BW}{2}\right)$ representing the lengths of the other two sides.

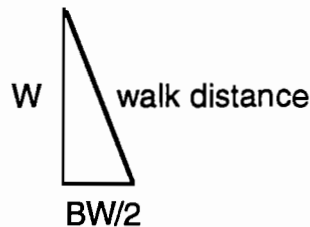


Figure 4-3 Weighted Average Walk Distance

Hence, the hypotenuse indicates an approximate weighted average walking distance. With this walk distance formula, the W -value can be altered to reflect effects of population density along the line or other socioeconomic variables. Both W and BW are held constant per station spacing isolating changes in total user cost due to in-vehicle travel time variations. Ingress and egress times are constant per station spacing when the W -value represents the actual travel distance and, therefore, any variance in travel time is felt in the in-vehicle time component.

Once the walk distance is known, it is divided by the walk rate (W_r) to obtain the walk time to or from the deboarding point. The unit of the rate is feet per second (ft/sec), so the resulting walk time is in seconds.

Time Calculated With W As Weighted Average Travel Distance

$$\text{walk time} = \frac{\sqrt{W^2 + \left(\frac{BW}{2}\right)^2}}{W_r}$$

Time Calculated With W As Actual Travel Distance

$$\text{walk time} = \frac{W}{W_r}$$

time in minutes

Discrete System

The discrete system permits boarding and alighting only at terminals. Figure 4-4 shows the catchment area for the walk access mode. Its width is the station spacing (S). The terminal (depicted by the small rectangle in the figure) lies in the middle of the area. The W -value again represents the distance people are willing to walk to access the system and is measured from the route on both sides and perpendicular to it. The spacing parameter (S) reflects the change in user time and cost as the spacing and the ensuing distance to travel parallel to the route to reach the terminal changes while the perpendicular travel distance (W -value) remains constant for each mode (Ref. 18, 33).

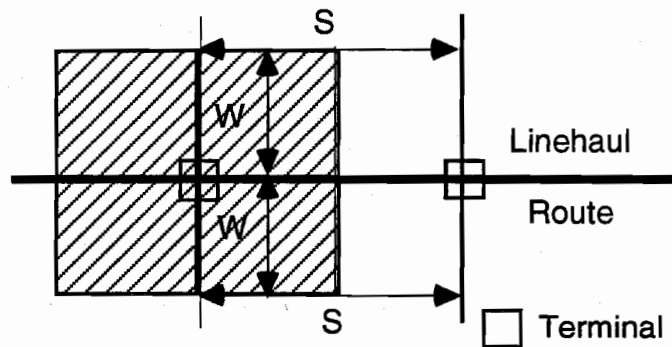


Figure 4-4 Walk Catchment Area-Discrete System

The next formula calculates the approximate weighted average walk distance based on the W -value and station spacing (S). The result is also in units of feet.

$$\text{walk distance} = \sqrt{W^2 + \left(\frac{S}{2}\right)^2}$$

distance in feet

The walk distance will be longer than the W -value for increasing station spacings because very few origins or destinations will be exactly perpendicular to the terminal or even close to it. Unlike the W -value, the spacing parameter (S) is dynamic and accounts for the change in the ingress and egress time for the

different interstation distances being tested. The time required to walk this distance is determined with the following formulas.

Time Calculated With W As Weighted Average Travel Distance

$$\text{walk time} = \frac{\sqrt{W^2 + \left(\frac{S}{2}\right)^2}}{W_r}$$

Time Calculated With W As Actual Travel Distance

$$\text{walk time} = \frac{W}{W_r}$$

time in minutes

If the walk rate W^2 is in feet/second units and distance in feet, the resulting time unit is seconds. The algorithm converts this to minutes.

The walk rates vary with age, trip purpose, and geographic location. The typical range extends from a 3.25 feet per second (ft/sec) value among the elderly up to a rate of 5.3 feet per second (ft/sec) for children. The remaining adults, furthermore, fall in the middle with average rates of 4.5 feet per second (ft/sec) to 4.8 feet per second (ft/sec) in the central business district (Ref. 9, 11).

Automobile As The Primary Access Mode

Driving an automobile or being dropped off from one is an ingress mode for all of the linehaul modes except local bus. Unlike the walk mode, the catchment area for driving is not unique for each operating policy. It is in the shape of a funnel (Ref. 17, 62). As Figure 4-5 depicts, the terminal facility (the parking lot and boarding platform represented by the rectangle) is at the downstream end of the area toward the work destinations. The tendency to avoid driving in the opposite direction away from the destination (upstream) defines the area as it is. The dimensions are quantified in terms of the number of minutes required to drive from the origin to the parking area and vice versa. The tendency to use the transit system rapidly diminishes beyond a fifteen (15) minute drive regardless of the distance or linehaul mode (Ref. 3, 29). So, the edge of the catchment area is fifteen (15) minutes upstream from the terminal.

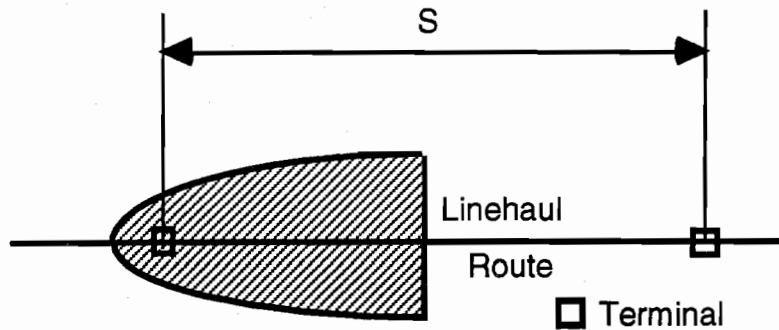


Figure 4-5 Auto Catchment Area

As the subsequent equation explains, the time to walk to the vehicle (T_V) and then to the platform from the parking space (T_P) plus the driving time (T_D) is the ingress time, or cost, to use this mode. The second element (T_P) is not included if the passenger is dropped off by another driver.

$$\text{drive time} = T_V + T_D + T_P$$

time in minutes

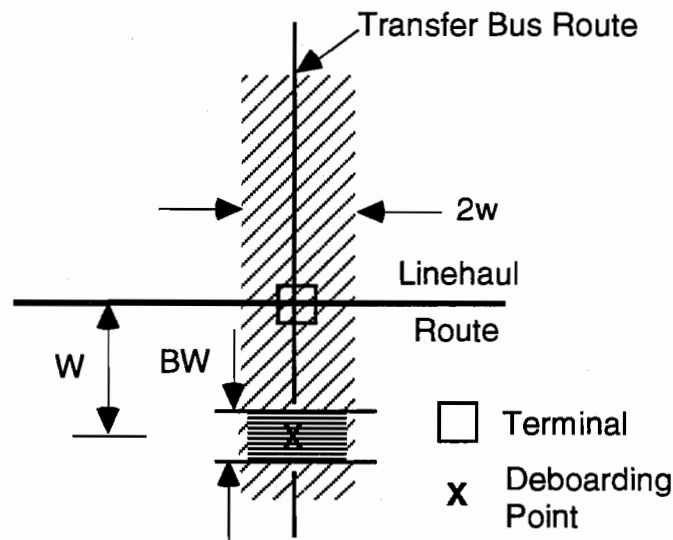
There are no differences among W -values for this mode because the algorithm requires driving and walking times in minutes as inputs and, hence, does not compute a travel time based on ingress distance and speed. The in-vehicle time in this case will reflect changes in user cost per station spacing for a particular weighted driving time. The driving time can be modified to represent different station spacings since the parameter can reflect a weighted average driving time or maximum desirable driving time. Ten (10) minutes for driving and two (2) for walking are typical average values.

Local Bus As The Primary Access Mode

Local bus mode is used to ingress the bus rapid, light rail, and rapid rail modes. Moreover, this mode is commonly used to egress all four linehaul modes. When only a portion of the journey can be made on the linehaul local bus, a transfer is made to another local bus which is considered to be the primary egress mode in this algorithm. Local bus is called a collector/distributor, or feeder, bus in the suburbs if it circulates through the local street system specifically to transport passengers to a linehaul terminal. The egress routes are typically established local bus routes in the central business district. A secondary mode, walking, is used to ingress and egress this primary access mode. With the exception of walking and driving to a linehaul terminal, all mode transfers are assumed to take place at a common point and, therefore, a distance and time to travel to another terminal are not included in the ensuing user time and cost calculations.

Continuous System

As stated previously, terminal catchment areas for walk access are rectangles, but when local bus is the primary access mode, the shape of the catchment area around the linehaul terminal is a distance only, simulated by the parameter W . Figure 4-6 depicts this scenario. The W -value again determines the travel distance aboard the transfer bus to the linehaul route. The parameter can, as usual, represent the maximum distance that people are willing to travel on the transfer route or it can represent an average trip length since one is not calculated as in the walk access mode. On the other hand, the parameter specifies the actual trip distance on the transfer bus if the W -value is to represent the actual distance. The distance is measured perpendicular to the transfer point on both sides. Since the algorithm assumes that a transfer of this type will be made where two routes intersect, the terminal spacing of the linehaul route is not a dimension used to demarcate the catchment area. The W -value is input in miles and the algorithm converts it to feet to maintain consistency among units.



**Figure 4-6 Local Bus Egress Catchment Area
Continuous System**

Furthermore, the local bus route itself has a catchment area similar to the one described for the walk access mode. Walking is a secondary access mode in this case since it is the mode used to ingress and egress the primary access mode. The area through which users walk to the bus terminal is also shown in Figure 4-6. The w -value is either the maximum acceptable walk distance or the actual distance walked to reach the terminal. The bandwidth is the other dimension of the catchment area and is used instead of the interstation spacing on the transfer route because of the operating policy. The algorithm assumes that the

transfer is made in the central business district (CBD) in order to reach a CBD destination. Thus, the same stopping policy is in effect for the transfer bus and the linehaul bus. The units are once again feet for the walk distance parameter. The following equation calculates the maximum walk distance over the catchment area in feet to reach the bus terminal.

$$\text{walk access distance} = \sqrt{(w)^2 + \left(\frac{BW}{2}\right)^2}$$

distance in feet

Once the egress distances are calculated for the bus ride and walk, they are divided by their respective egress rates to obtain the time required to ride the bus and walk to the destination point.

Time Calculated With W As Weighted Average Travel Distance

$$\text{local bus egress time} = \sqrt{HW} + \left(\frac{W}{(AOS/60)}\right) + \left(\frac{\left(\sqrt{w^2 + (BW/2)^2}\right)}{(W_r * 60)}\right)$$

Time Calculated With W As Actual Travel Distance

$$\text{local bus egress time} = \sqrt{HW} + \left(\frac{W}{(AOS/60)}\right) + \left(\frac{w}{(W_r * 60)}\right)$$

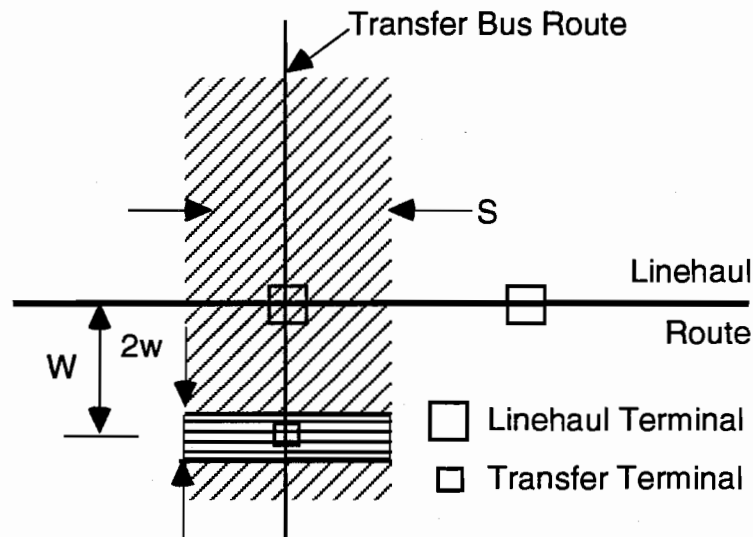
HW = headway in minutes
time in minutes

As the above formulas show, the time to wait to board the transfer bus is also a component of the access time. The square root of the headway (*HW*) (fifteen (15) minutes is a representative value) is a typical predictor of waiting time for routes with headways less than thirty (30) minutes (Ref. 15). The algorithm is not concerned with the interstation spacing of the access mode since users tend to be more concerned with the linehaul mode they are trying to ingress or egress and will not use an access mode if it takes too much time relative to the time spent riding on the linehaul mode. Therefore, an overall average bus operating speed (*AOS*) is sufficient to calculate the in-vehicle time. This value is usually known in miles per hour (mph), so the algorithm accepts this and converts the number to units of feet per minute (ft/min) for the travel time calculation. The algorithm provides a default value of only twelve (12) miles per hour (mph) since there will be many stops with this type of operation. An average *W*-value is two (2) or three (3) miles. Finally, for the walk time, the walk rate is multiplied by sixty (60) to convert it to feet per minute (ft/min) and then is divided into the walk distance in feet. These default values are identical to those

previously mentioned for the walk access mode. The egress time for the local bus mode is the sum of these three user cost components.

Discrete System

The dimensions of the catchment area are different for the discrete system (second operating policy) when local bus is used to egress from local bus, bus rapid, light rail, or rapid rail routes. As Figure 4-7 shows, the width of the catchment area for the egress bus itself is a fraction of the linehaul interstation spacing in the central business district. The w -value for this catchment area is measured along the route length. The former dimension reflects the effect of the linehaul station spacing on the users' secondary ingress/egress cost since all egress routes presumably intersect with the linehaul route at a common point (the linehaul terminal). The w -value represents the maximum desirable walk travel distance along the route to reach the egress bus terminal. The W -value is the same and can stand for the maximum acceptable egress bus ride or the average bus trip length. These values can also represent the actual walk and ride distances if this version of the method is chosen. The equations for the egress time via local bus follow the figure.



**Figure 4-7 Local Bus Egress Catchment Area
Discrete System**

Time Calculated With W As Weighted Average Travel Distance

$$\text{local bus egress time} = \sqrt{HW} + \left(\frac{W}{(A \cdot 60)} \right) + \left(\frac{(\sqrt{w^2 + (\frac{S}{2})^2})}{(W_r \cdot 60)} \right)$$

Time Calculated With W As Actual Travel Distance

$$\text{local bus egress time} = \sqrt{HW} + \left(\frac{W}{(A \cdot 60)} \right) + \left(\frac{w}{(W_r \cdot 60)} \right)$$

HW = headway in minutes
time in minutes

There is yet another catchment area around the bus rapid, light rail, and rapid rail terminals if the ingress mode is local bus (second and third operating policies). The area portrayed in Figure 4-8 indicates that the spacing of the linehaul terminal has added a width dimension to the linehaul catchment area that the previous areas do not have. This is due to the fact that the ingress bus will circulate through the local street system to collect passengers for their commute to work on the linehaul mode and will not necessarily follow a one-directional route. The linehaul terminal is in the middle of the influence area that is parallel to the route while the *W*-value represents the maximum acceptable linehaul route distance. The spacing parameter (*S*) directly affects the user cost by affecting the access distance.

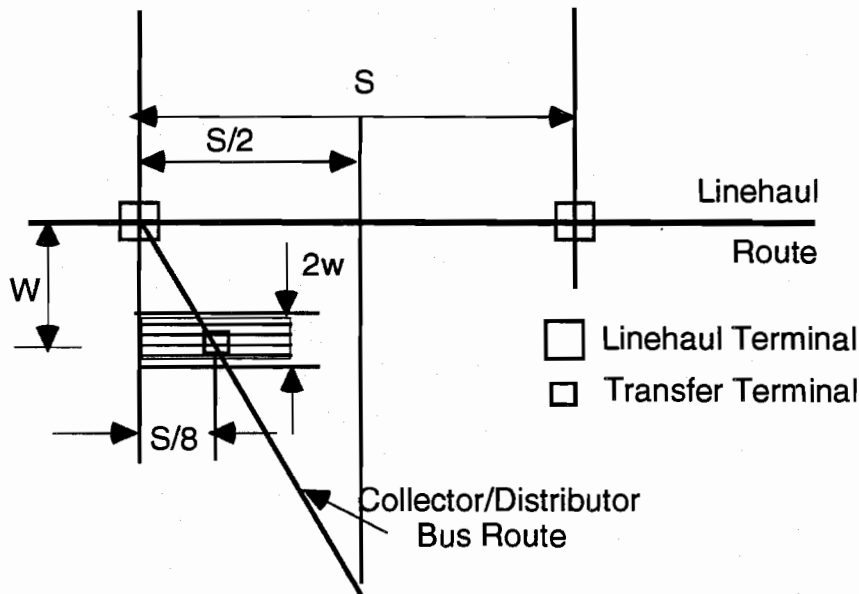


Figure 4-8 Local Bus Ingress Catchment Area

The catchment area for ingress is similar to that for egress except that the perpendicular component of the preferred walking distance ($S/8$) is an even smaller fraction of the linehaul terminal spacing since this spacing is presumably much greater in the ingress area (suburbs). This walk distance is calculated as follows.

$$\text{walk distance} = \sqrt{(w)^2 + \left(\frac{s}{8}\right)^2}$$

distance in feet

Units for these parameters and their typical values are the same as those previously mentioned for the walk mode. Moreover, the square root of the headway is again used to calculate the wait time. The following equations incorporate these parameters to calculate the user ingress time and, hence, cost based on station spacing.

Time Calculated With W As Weighted Average Travel Distance

$$\text{local bus ingress time} = \left(\frac{\left(\sqrt{w^2 + \left(\frac{s}{8}\right)^2} \right)}{(w_r * 60)} \right) + \sqrt{HW} + \left(\frac{\left(\sqrt{w^2 + \left(\frac{s}{8}\right)^2} \right)}{(AOS/60)} \right)$$

Time Calculated With W As Actual Travel Distance

$$\text{local bus ingress time} = \left(\frac{w}{(w_r * 60)} \right) + \sqrt{HW} + \left(\frac{w}{(AOS/60)} \right)$$

HW = headway in minutes

time in minutes

Light Rail As The Primary Access Mode

The final access mode is light rail and it is used to ingress and egress bus, rail rapid and other light rail routes. The algorithm assumes that the light rail ingress route will run on arterial or collector streets perpendicular to the linehaul route, so the distance traveled on the light rail collector is W (Figure 4-9). The figure also includes the catchment areas of the three secondary access modes (walk, auto, and local bus) to the light rail terminals. The spacing of the light rail or secondary local bus terminals is again of no importance to this methodology, so the in-vehicle times for them will be calculated with the distance and

average operating speed (AOS). The parameters have the same meanings, units, and representative values as in the previous discussions. The walk distances expressed as fractions of the linehaul spacing decrease as their percentage of the overall trip length decreases. They reflect the time required to access the transit system for each particular station spacing. The W and w values can also be altered to reflect the effect that acceptable access distances have on station spacing. The following equations compute the ingress time for the secondary access modes and the primary access mode. A realistic percent of the modal split captured by each is in parentheses next to the equation. The secondary travel times are each multiplied by their respective modal split percentages to obtain a weighted average light rail ingress time and are added to the wait and in-vehicle time.

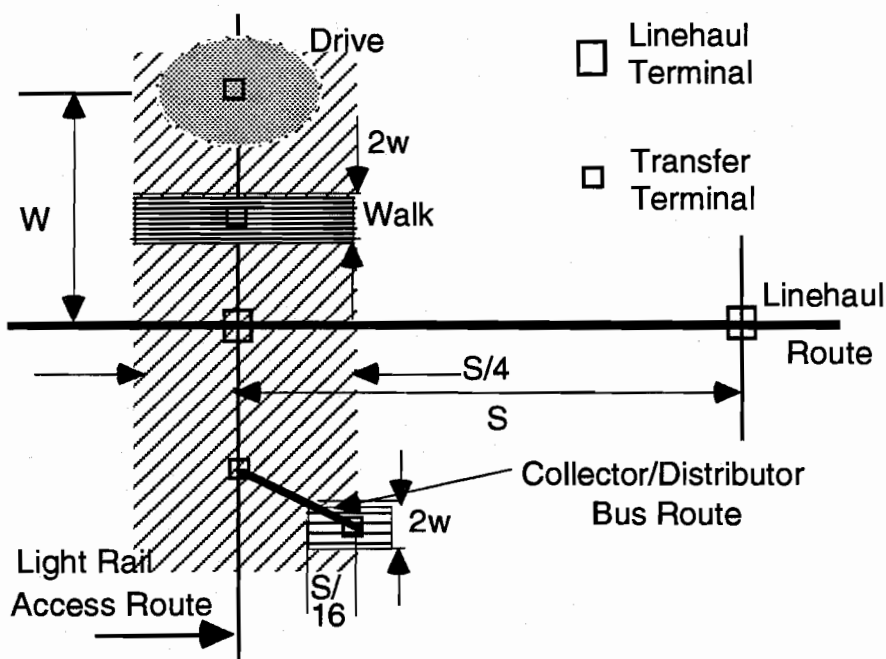


Figure 4-9 Light Rail Ingress Catchment Area

Time Calculated With W As Weighted Average Travel Distance

$$\text{walk time} = \frac{\sqrt{(w)^2 + \left(\frac{s}{8}\right)^2}}{(W_r * 60)} \quad (40\%)$$

$$\text{drive time} = T_v + T_D + T_p \quad (40\%)$$

$$\text{local bus time} = \left(\frac{\left(\sqrt{w^2 + \left(\frac{s}{16}\right)^2} \right)}{(W_r * 60)} \right) + \sqrt{HW} + \left(\frac{\left(\sqrt{w^2 + \left(\frac{s}{8}\right)^2} \right)}{(AOS/60)} \right) \quad (20\%)$$

$$\text{light rail time} = \sqrt{HW} + \left(\frac{w}{(AOS/60)} \right)$$

$$\text{total ingress time} = .40(\text{walk}) + .40(\text{drive}) + .20(\text{localbus}) + 1.0(\text{lightrail})$$

Time Calculated With W As Actual Travel Distance

$$\text{walk time} = \frac{w}{(W_r * 60)} \quad (40\%)$$

$$\text{drive time} = T_v + T_D + T_p \quad (40\%)$$

$$\text{local bus time} = \left(\frac{w}{(W_r * 60)} \right) + \sqrt{HW} + \left(\frac{w}{(AOS/60)} \right) \quad (20\%)$$

$$\text{light rail time} = \sqrt{HW} + \left(\frac{w}{(AOS/60)} \right)$$

$$\text{total ingress time} = .40(\text{walk}) + .40(\text{drive}) + .20(\text{localbus}) + 1.0(\text{lightrail})$$

HW = headway in minutes
time in minutes

The catchment area around the linehaul terminal for the egress light rail route is shown in Figure 4-10. It is similar to the shape of the light rail ingress catchment area. The bandwidth parameter (BW) replaces the spacing parameter (S) if the third operating policy is in effect. It is again assumed that the transfer will take place at a single multi-modal interface terminal, so the primary egress catchment area is has no width dimension. The secondary egress mode is walking. Since the destination is likely in the central business district, the algorithm presumes that a secondary egress mode from the light rail route (local bus) will not be required to complete the journey (Ref. 16, LR7). The spacing parameter can again be altered to examine the effect of the interstation spacing on user time. The W and w variables represent the same cost parameters as in previous discussions. The following equations compute the egress travel time via light rail.

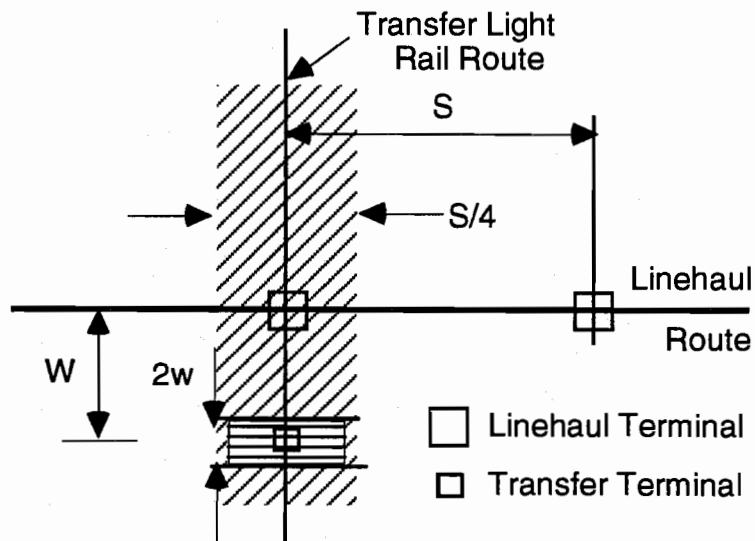


Figure 4-10 Light Rail Egress Catchment Area

Time Calculated With W As Weighted Average Travel Distance

$$\text{light rail egress time} = \sqrt{HW} + \left(\frac{W}{(LOS/60)} \right) + \left(\frac{\left(\sqrt{W^2 + \left(\frac{S}{8} \right)^2} \right)}{(W_r * 60)} \right)$$

Time Calculated With W As Actual Travel Distance

$$\text{light rail egress time} = \sqrt{HW} + \left(\frac{W}{(LOS/60)} \right) + \left(\frac{W}{(W_r * 60)} \right)$$

HW = headway in minutes

time in minutes

With the exception of local bus ingress, most users have a choice of available access modes and, therefore, a combination is used to travel to and from the linehaul terminal. This algorithm recognizes the occurrence of this modal split and has provisions to deal with it. The percentage of users captured by each mode is input and then multiplied by the ingress and egress travel time for that particular mode. The resulting values are added as in Table 4-4 to obtain an overall weighted time per station spacing.

Table 4-4 Weighted Travel Time Per Station Spacing

$$\begin{aligned} \text{weighted travel time} = & (\%walk)(\text{walk time}) + (\%auto)(\text{drive time}) + \\ & (\%bus)(\text{bus ride time}) + (\%rail)(\text{rail ride time}) \end{aligned}$$

Any combination of modes and modal splits can be tested with this format, thus providing flexibility. The percent of riders using a particular access mode varies depending on the linehaul mode. The algorithm includes the percentages associated with the modal splits experienced by existing systems as a guide. Table 4-5 lists these for each linehaul mode.

Table 4-5 Access Modal Split Percents

Activity	Linehaul Mode			
	Local Bus	Bus Rapid	Light Rail	Rapid Rail
<u>Ingress %</u>				
Walk	100	10	15	23
Auto	0	65	65	55
Local Bus	0	20	20	20
Light Rail	0	5	0	2
<u>Egress %</u>				
Walk	70	65	65	65
Local Bus	30	30	30	30
Light Rail	0	5	5	5

The ingress and egress times are two of the three out-of-vehicle costs incurred by transit riders.

4.1.4 Wait Time Parameter

The time to wait at the terminal for the linehaul vehicle is the second out-of-vehicle time component. It is measured from the time the user reaches the terminal to the time the vehicle is boarded. This parameter is highly variable since it depends on user preferences and trip purposes. For example, commuters who ride the same route at the same time everyday are familiar with the arrival time of the vehicle and minimize their wait time based on this knowledge. Shoppers, on the other hand, may not be as familiar with the arrival tendencies of the vehicle at a particular terminal and will depend on the scheduled arrival time to plan their wait. The route headway (number of minutes between successive arrivals at a terminal) is the parameter used to predict the passenger wait time (Ref. 2, 11). For instance, a passenger will likely plan to wait longer for a vehicle with a lengthy headway because the consequence of missing it is an increased travel time and a correspondingly higher cost. Missing a vehicle with a shorter headway, on the contrary, poses less time-loss risk. The algorithm computes the passenger wait time in a variety of ways to account for this variability. The square root of the headway or one-half the headway are two of them. Another option is to perform either operation depending on the length of the headway itself. The former generally applies when the headway is less than thirty (30) minutes and the latter when it is

greater (Ref. 15). The final option is to input an exact waiting time. The headway (or exact waiting time) is input in minutes and the resulting unit is minutes. Table 4-6 lists the options in their mathematical form. The wait time is added to the ingress and egress time to obtain the total user out-of-vehicle time and cost.

Table 4-6 Wait Time Calculation Options

$$\sqrt{HW}$$

$$.5(HW)$$

$$\sqrt{HW} \text{ if } HW < 30; .5(HW) \text{ if } HW \geq 30$$

Input exact time

(HW = headway in minutes)

4.2 Linehaul Time Parameters

The second element of travel time is in-vehicle riding time. As previously mentioned, this time is very dependent on station spacing or bandwidth and is calculated in the same manner regardless of the operating policy type. The number of vehicle cycles the user rides through between the ingress and egress terminals dictates this part of the overall travel time. The four components of the vehicle cycle as shown in Figure 3-6 are the acceleration, cruise, deceleration, and dwell times. Table 4-7 explains how they determine the cycle time. This cycle time is multiplied by a parameter (J) to determine the linehaul travel time. J is the number of cycles for either the entire route length or the average trip length. These distances are divided by the interstation spacing to calculate J .

Table 4-7 In-Vehicle Time Calculation

$$\text{cycle time} = (\text{accel} + \text{cruise} + \text{decel} + \text{dwell})$$

$$\text{linehaul time} = J(\text{cycle time})$$

$$J = \frac{\text{route length or average trip length}}{S}$$

If the result is not an integer, then the remaining fraction is converted to the physical distance it represents and is tested to see if the desired cruising speed can be reached.

The algorithm approximates constant acceleration and deceleration rates (Ref. 13, 399). An average acceleration rate (a) for the vehicle based upon passenger comfort provides the input for the calculation of the time spent accelerating. An average deceleration rate (d) obtained in the same way is used to figure the third cycle time component. These values can be procured from the vehicle manufacturer and entered in the equations with units of feet per second squared (ft/sec^2). Typical rates range from 2.3 feet per second squared (ft/sec^2) for buses to 5.25 feet per second squared (ft/sec^2) for rail vehicles (Ref. 27, 227 and 346). The subsequent equations calculate the time required to accelerate to the desired cruising speed (V) and to decelerate from this speed to zero velocity. This velocity can be based on the posted speed limit over the route or on an established policy if no speed limit governs the route. It can be one of the parameters that accounts for the travel time variance in the continuous system if it changes in each route segment. The algorithm takes the velocity in feet per second (ft/sec) to maintain unit consistency.

$$\begin{aligned} \text{time to accelerate} &= \frac{V}{a} \\ \text{time to decelerate} &= \frac{V}{d} \end{aligned}$$

The distance covered while accelerating and decelerating is also a function of the desired cruising velocity.

$$\begin{aligned} \text{distance to accelerate} &= \frac{V^2}{2a} \\ \text{distance to decelerate} &= \frac{V^2}{2d} \end{aligned}$$

The time spent traveling at the desired cruising velocity (V) is a function of the station spacing (S). The cruise distance is the distance between the point at which the cruise speed is reached and the point at which the vehicle must begin decelerating in order to stop at the terminal. These three distances sum to equal the distance between terminals. The time spent cruising results when the cruise distance is

divided by the velocity (V). The formula for this procedure with S representing the interstation spacing follows.

$$\text{cruise time} = \left(S - \left(\frac{V^2}{2a} \right) - \left(\frac{V^2}{2d} \right) \right) / V$$

There are three possible results for this equation. First, if the cruise distance is negative, the spacing is not great enough for the vehicle to reach the desired velocity. These two items are unknown in this case because the velocity is used to calculate the time and is not known if the desired speed is not reached. Hence, the algorithm computes the ratio of the deceleration rate to the acceleration rate to express the former in terms of the latter and produce only one unknown. The following is the expression for this ratio with x representing the fraction of the acceleration rate that is the deceleration rate.

$$x(a) = \frac{d}{a}$$

The next equation is another way to calculate the acceleration and deceleration distances, but uses time (t) instead of velocity.

$$\text{distance to accel and decel} = \frac{1}{2} at^2 + \frac{1}{2} dt^2$$

Since the cruise speed is never reached, this distance is the spacing between terminals. Hence, the equation with a single unknown looks like the following.

$$S = \frac{1}{2} at^2 + \frac{1}{2} (xa)t^2$$

Solving for the time (t) yields the next equation. The 60 converts time from seconds to minutes.

$$t = \frac{\left(\sqrt{\frac{2S}{a(1+x)}} \right)}{60}$$

This time is added to dwell time to obtain cycle time and then multiplied by J to yield total full cycle time. A fractional spacing is not tested in this case since cruise velocity (V) would not be reached within this distance if it could not be reached within the full spacing. Therefore, the fractional distance is assumed to be spent accelerating and decelerating and the time involved is figured in the same way as for the full cycle. This time will be a very minimal part of the cycle time. It is added to the full cycle time to obtain the in-

vehicle riding time per station spacing. Figure 4-11 portrays this regime in which the desired cruise velocity is not reached and the acceleration and deceleration rates are not equal.

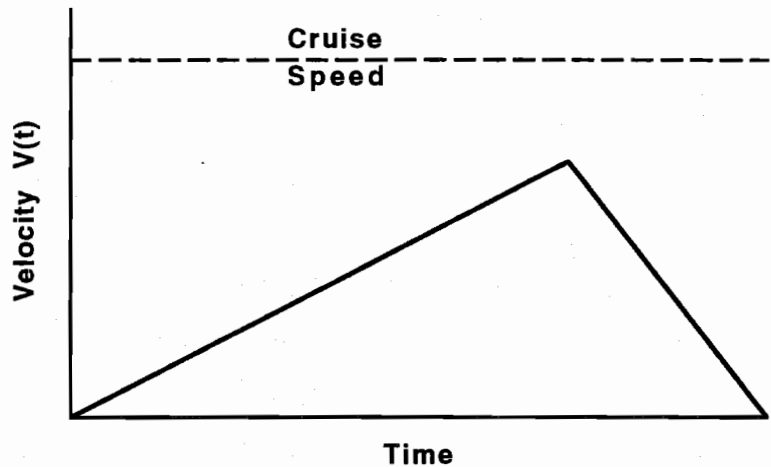


Figure 4-11 Desired Cruise Velocity Is Not Reached

A second possible result is for the spacing to equal the sum of the distance to accelerate and decelerate. The vehicle will accelerate fully to the desired cruising velocity and then immediately begin decelerating to stop at the next terminal (Figure 4-12). The following formula calculates the first two components of the complete cycle for this case.

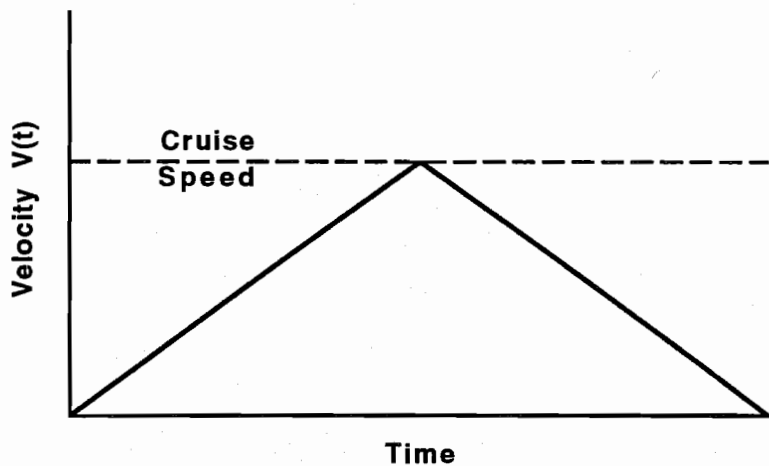


Figure 4-12 Desired Cruise Velocity Is Reached

$$\text{time to accel and decel} = \frac{V}{a} + \frac{V}{d}$$

Once again, the algorithm does not test the fractional spacing to see if cruise speed can be reached because no time is spent actually cruising and, hence, the entire distance is spent accelerating and decelerating. So, the time for this fractional spacing is calculated as before with the ratio of the deceleration rate to the acceleration rate. The total time to be added to the dwell time is the sum of the time for the complete cycles and the fractional cycle.

Finally, if the cruise distance is greater than the terminal spacing (S), then it is divided by the cruise velocity to obtain the time to cover one vehicle cycle (Figure 3-6). The mantissa (y) of the J value is multiplied by the spacing and compared to the acceleration plus deceleration distance to test if the cruising speed can be reached within the fractional spacing.

$$y(S) > \left(\left(\frac{v^2}{2a} \right) + \left(\frac{v^2}{2d} \right) \right)$$

$$y(S) = \left(\left(\frac{v^2}{2a} \right) + \left(\frac{v^2}{2d} \right) \right)$$

$$y(S) < \left(\left(\frac{v^2}{2a} \right) + \left(\frac{v^2}{2d} \right) \right)$$

The same three situations can result from this operation. The same procedure is followed for each case and the fractional time is added to the time for complete cycles to determine the total user in-vehicle time and cost.

Dwell time is stationary time at the terminal loading and unloading passengers and is the fourth vehicle cycle time constituent. It is another cost parameter that is highly variable (with the exception of automated rapid rail systems) and has an influence on system performance. The number of passengers boarding and alighting, simultaneous or individual occurrence of these two activities, number of doors used, time of day, fare payment method, and physical capabilities of passengers are all factors that contribute to the dwell time variability among stops (Ref. 7, 10). Of these factors, the first is the only one that will be common to all transit systems and will, therefore, be used to calculate dwell time. This algorithm provides an option of using a Poisson-based procedure to predict total dwell time or entering an exact dwell time in seconds.

Since passenger arrivals at terminals are random and independent, the procedure assumes a Poisson arrival distribution and determines dwell time by calculating the actual number of stops made based on boarding or deboarding demand and the number of passengers needing service. Demand can be predicted with standard forecasting techniques. Dividing route length by spacing determines possible stops (N). It is the same formula as the one that determines number of cycles since it is assumed that one cycle is necessary to travel between the last and first terminal.

$$N = \frac{\text{route length or trip length}}{S}$$

The integer of the quotient is used. The boarding or deboarding demand per minute (pax/min) is divided by this number of stops to determine arrivals per minute at the stop (X_i) or the departures per minute from the vehicle (Y_i). When these values are changed for each route segment the variance in travel time for the continuous system occurs.

$$X_i = \frac{\text{board pax}}{\text{min}} \quad \text{or} \quad Y_i = \frac{\text{deboard pax}}{\text{min}}$$

The probability (P) of a system user boarding or deboarding at a terminal (necessitating a dwell time) is calculated as follows:

$$P(\text{at least one board or alight}) = 1 - e^{-(X_i + Y_i)HW}$$

HW is the route headway in minutes. This result is multiplied by the possible stops (N) to obtain the number of stops actually made (A).

$$A = P * N$$

The following formulas for the mean arrival or departure rate (m) in passengers per minute (pax/min) determine the number of passengers at these actual stops.

$$m = X_i H \quad \text{or} \quad m = Y_i H$$

This rate is inserted into the Poisson probability distribution.

$$P(x) = \frac{m^x e^{-m}}{x!}$$

The x value represents the number of passengers and the formula calculates the probability of this number actually boarding or alighting the terminal. The desired probability for accuracy is input into the algorithm and the Poisson probabilities are summed until they equal or exceed this percent. Q represents the number of passengers that will yield the desired percent.

$$\sum_{x=0}^Q P(x) \geq (\text{desired}\%)$$

The number of passengers per stop (pax/stop), Q , is multiplied by the actual number of stops made (A) and the passenger boarding headway (HW_b) to obtain the total loading time (T_L) required.

$$\text{loading time} = Q * A * HW_b$$

The passenger headway is input in seconds (typically two (2)) and the result is in seconds. The same procedure determines the time needed to unload (T_D) the transit riders. Q in this case represents the number of deboarding passengers per stop (pax/stop) and a typical deboarding headway (HW_d) is again two (2) seconds.

$$\text{unloading time} = Q * A * HW_d$$

The final step in this process is to account for the time spent initiating these activities. Typical values are four (4) seconds for the boarding start up time (I_b) and two and one-half (2.5) seconds for the deboarding start up time (I_d). The algorithm calculates the total boarding (T_B) and deboarding (T_A) times with the following formulas and converts the result to minutes.

$$\text{total boarding time} = (A * I_b) + T_L$$

$$\text{total deboarding time} = (A * I_d) + T_D$$

If these two activities occur simultaneously, the algorithm selects the largest value. Otherwise, the times are added together. This time is divided by the number of stops made (A) to get the dwell time per stop.

$$\text{dwell time per stop} = \frac{\max(T_B, T_A)}{A}$$

The resulting dwell time is added to acceleration, cruise, and deceleration times to compute cycle time.

The four components of the vehicle cycle time are calculated separately and added together to obtain the full cycle time and the fractional cycle time. These are summed to determine the in-vehicle time. Two in-vehicle times are calculated in this algorithm: the time for an average trip length and the time to complete one route pass. The station spacing has a significant effect on this portion of the user travel time and cost.

4.3 Summary

Out of vehicle time spent accessing the transit system and time spent riding in the vehicle are the two user travel time components. Previous discussions broke these two times down to the individual parameters. Adding these elements yields total user travel time per a given station spacing. With all available options, this algorithm provides numerous ways to calculate this time. There are four linehaul modes and several possible ingress and egress mode combinations. Any of four methods can be used to account for waiting time. The dwell time can be input or predicted using Poisson and every possible scenario for travel time between terminals is addressed. The station spacing affects each of these parameters in some way and all are necessary in a terminal spacing analysis. The travel time is computed for a range of possible station spacings by changing the spacing (S) or bandwidth (BW) variable. The shortest of these times equates to the least user cost and the corresponding spacing is the optimal transit terminal interstation spacing.

CHAPTER 5

THE COMPUTERIZED MATHEMATICAL ALGORITHM

Calculation of cost (access and linehaul time) to use public transportation requires inclusion of several parameters. The station spacing has a direct effect on many of them and, consequently, the overall trip time. In addition, as the previous chapter shows, some are highly variable since they represent human behavioral characteristics and tendencies. As such, each must be examined over a range of values. Furthermore, groups of variables should be altered in different combinations to create a multitude of possible user preferences. These procedures are performed for each station spacing and are repeated several times depending on the possible spacing range. Thus, an analysis of optimal interstation terminal spacing performed by hand is quite tedious and the possibility of error is great. To alleviate this situation, the algorithm and all of its equations and variables has been embodied within a computer program.

The algorithm has been computerized with the Microsoft Excel™ (Version 4.0) program. The code for it was written on macro sheets because they customize the capabilities of Excel so it can perform specialized tasks. The algorithm itself is contained in a series of formulas that instruct the macro program to calculate travel time per station spacing. The formulas use functions unique to macros to obtain required inputs, calculate travel time, control the loop to repeat calculations over the range of possible station spacings, and to direct the program between all these activities. As Appendix A demonstrates, the typical macro is built with two columns. The formulas are written in the right column while the left column provides a code explanation.

Excel was chosen for this task for two reasons. First is its user-interface capability. The algorithm is very input-intensive and customized applications of Excel features provide a non-intimidating format with which to prompt the program user for necessary information. These applications also serve as a mechanism to include representative values of user time parameters as a guide for the program user. Moreover, Excel can package algorithm results in tabular and graphical form, both of which are easy to comprehend and interpret. The figures in the third chapter were created in Excel. Results in this format will be convenient to demonstrate and explain to citizens why the chosen terminal spacing is the optimal one for their transit system.

5.1 Program Components

Several macros were created to analyze transit terminal spacing with this algorithm. They appear in Appendix A. One main routine titled *Backbone* controls the program and directs it between the subroutines listed in Table 5-1. It also opens and closes the appropriate macro files as needed. These files are protected and, therefore, can not be changed and saved under the same name. This feature is useful to preserve the integrity of the algorithm since altering a file can render the results inaccurate.

Table 5-1 Program Subroutines

Summcalc
Typeone
Typetwo
Railtwo
Typethree
Linehaul
Dwell

Backbone also prompts the user for a file name under which to save the summary statistics and saves them. It saves the results at the end of the program with a name provided from the subroutines. The tasks accomplished by the *Backbone* routine are listed in Table 5-2.

Table 5-2 Backbone Tasks

Direct program between subroutines
Open and close files
Save summary and results

5.1.1 Program Subroutines

The *Summcalc* subroutine opens the file that contains all inputs from a previous analysis of optimal interstation spacing and presents it to the user on a worksheet. Part of the worksheet is shown in Figure 5-1.

As Table 5-3 explains, the user can change any values on this sheet and travel time per station spacing is recalculated and the results tabulated and graphed within this subroutine. This is a quick way to study the effect of changing an individual parameter or one group of them since the lengthy standard input process

OPERATING POLICY:	Two
LINE HAUL MODE:	Bus Rapid
SPACING:	INCREMENT:
W PREFERENCE:	Actual Distance Traveled
ACCESS MODES:	Walk
	Auto
	Local Bus
WAIT TIME (min):	2
POISSON INPUTS:	BOARD DEMAND (pax/hr):
	BOARD PAX START UP TIME (sec):
SEGMENTS:	SEGMENT LENGTH (mi):

Figure 5-1 Summary Sheet

Table 5-3 Summcalc Tasks

Present inputs from a previous run
Allow changes of parameter values
Recalculate travel time
Tabulate and graph results

is avoided. It is also more expedient for users who are very familiar with the program and the required inputs and do not need to be stepped through the standard input process. Rather, they can go directly to the sheet with the summary statistics and enter the values of the travel time components and quickly obtain the time per station spacing results. The program is also much faster since the *Summcalc* file and the file containing the input summary are the only ones in addition to *Backbone* that have to be opened and closed in order to perform the analysis.

The *Typeone* subroutine houses the formulas that apply to the first operating policy (continuous system). As Table 5-4 shows, this macro begins by opening the pertinent files. These files produce versions of the Excel input applications which are customized for the input requirements of this operating policy. Each input is recorded on the summary worksheet once it has been keyed in. The macro then calculates the components of the user out-of-vehicle time (ingress, wait, and egress) for each possible

Table 5-4 Typeone Tasks

Open files pertinent to first policy
Obtain continuous system inputs
Record inputs on summary sheet
 Calculate access travel time
 Refer control to Linehaul
 Tabulate and graph results

of the user out-of-vehicle time (ingress, wait, and egress) for each possible station spacing. The next duty of this subroutine is to refer to the *Linehaul* macro so that the riding time on the local bus can be calculated for each spacing. Compiling the results and transferring the ensuing table and graph to a worksheet are the final tasks accomplished by the *Typeone* subroutine.

The formulas needed to study the optimal interstation spacing for a discrete system (second operating policy) are contained in the *Typetwo* subroutine. It performs the same functions as the *Typeone* subroutine (refer to Table 5-4) for the local bus and bus rapid linehaul modes. It refers control to the *Railtwo* macro, however, to gather the required inputs, calculate the travel time per station spacing, and assemble results for light and rapid rail modes. Each also calls the *Linehaul* routine at the appropriate place in the analysis sequence.

Table 5-5 Typetwo Tasks

Open files pertinent to second policy
 Obtain discrete system inputs
 Record inputs on summary sheet
 Calculate access travel time
 Refer control to Linehaul
 Tabulate and graph results
 Refer control to Railtwo if necessary

The *Typethree* subroutine addresses the third and final operating policy. As Table 5-6 indicates, it performs tasks similar to those of the *Typeone* subroutine. The customized input applications gather information needed to calculate travel time on a discrete and a continuous system. After computing out-of-vehicle and in-vehicle times to use the system for various station spacings and bandwidths, the formulas in this subroutine instruct the macro to package the results in a table and on a graph of the station spacing versus user travel time.

Table 5-6 Typethree Tasks

Open files pertinent to third policy
Obtain discrete and continuous system inputs
Record inputs on summary sheet
Calculate access travel time
Refer control to Linehaul
Tabulate and graph results

The *Linehaul* subroutine determines the in-vehicle component of the travel time and, thus, is composed of formulas that execute calculation of in-vehicle riding time per station spacing. Inputs for these calculations are obtained from the user and recorded on the summary sheet in the subroutine that passes control to *Linehaul*. The subroutine calculates acceleration, cruise, and deceleration times for each station spacing. This macro directs control to *Dwell* subroutine (as shown in Table 5-7) if the program user chooses to calculate the dwell time with the Poisson procedure. When the value is returned to *Linehaul*, it is added to the other three cycle time components. Otherwise, the exact dwell time is obtained from the operating policy subroutine that called the *Linehaul* subroutine. This macro is not responsible for opening any files or recording the values it uses in a summary sheet.

Table 5-7 Linehaul Tasks

Calculate linehaul travel time
Refer control to Dwell
Tabulate and graph results

5.1.2 Algorithm Inputs

The inputs to this program are the responses obtained from users when they are prompted for information. They serve the four purposes listed in Table 5-8. First, some inputs are requests for names under which to save the summary file and the results file created in each of the operating policy subroutines. The linehaul mode to be examined and the modes used to access it are other inputs the program gains from the user interface. Values for variables used by the equations in the previous chapter that calculate travel time for a given station spacing or bandwidth compose a significant majority of the needed inputs. Finally, the program uses inputs to determine which options the user wishes to exercise and then directs the macros accordingly. The inputs are obtained from the interface between the program and the user via input or dialog boxes. Both are standard Excel applications that have been customized for the optimal interstation spacing algorithm.

Table 5-8 Input Functions

Provide names to save files
Specify linehaul and access mode
Enter parameter values
Direct macro among chosen options

An input box is a simplified form of the dialog box and requests a single piece of user input. A formula specific to macros creates and displays the input box in one line of code (shown in Table 5-9). The first set of quotation marks contains the message specifying the requested input while the title for the box is in the second set. The number indicates whether the response should be in text or numeric only. This particular formula requires a response in the form of a number as opposed to text.

Table 5-9 Macro Input Formulas

**INPUT("Enter a name under which to save the
results from this run.",2,"RESULTS")
DIALOG.BOX(LHM1!B3:H22)**

Figure 5-2 is an example of the input box created by this formula. Furthermore, the second macro formula displays the dialog box, using one line of code, that asks for information about the linehaul mode for the first operating policy. The dialog box itself was created in a separate macro file with a dialog box definition table similar to the one in Figure 5-3. *LHM1* is the name the file is saved under and *B3:H22* is the dimension expression of the screen dialog box. The exclamation mark indicates to the subroutine macro that it has to look externally in another file for the dialog box definition table. The text requesting the input and title of the dialog box along with the form the input must be in are spelled out in the definition table. Figure 5-4 shows the dialog box as it appears to the user on the screen. Both forms of the input boxes clearly state the input being asked for along with the units (when appropriate) the macro assumes. Values for each input request have been built into each input and dialog box with the exception of the names to use to save summary and result files. These values appear in the boxes when the subroutine macro displays them and are algorithm default values. They are the representative values described for each input in the previous chapter and are user guides.

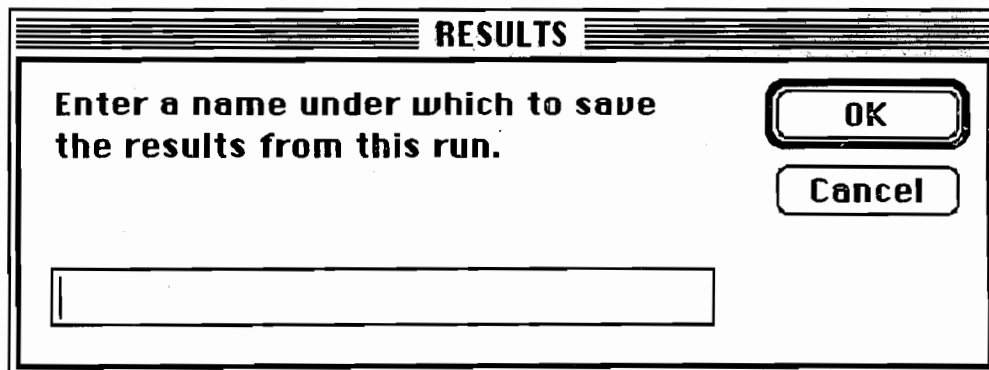


Figure 5-2 Results File Name Input Box

Hence, if program users have no information about the value of the input as it pertains to their system and system patron preferences, they can trust these defaults to be reasonable, average

parameter values. However, they are encouraged to enter values that will tailor the analysis specifically to their user requirements and needs, governing policies, and equipment. The input and dialog boxes are configured such that the same default values appear every time the box is displayed. The inputs entered in them are recorded to the summary sheet and used in subsequent calculations.

BOX NUMBER	NUM	H O R	V E R	TEXT	INIT RES
				LINE HAUL INFO	
STATIC TEXT	5			LINE HAUL MODE:	
TEXT EDIT BOX	6				Local Bus
STATIC TEXT	5			BANDWIDTH (ft):	
NUMBER EDIT BOX	8				1320
STATIC TEXT	5			ROUTE HEADWAY (min):	
NUMBER EDIT BOX	8				30
STATIC TEXT	5			ACCELERATION RATE (ft/sec ²):	
NUMBER EDIT BOX	8				2.62
STATIC TEXT	5	220	12	DECELERATION RATE (ft/sec ²):	
NUMBER EDIT BOX	8				13.12
STATIC TEXT	5			CRUISE VELOCITY (ft/sec):	
NUMBER EDIT BOX	8				44
STATIC TEXT	5			AVERAGE TRIP LENGTH (mi):	
NUMBER EDIT BOX	8				4.5
STATIC TEXT	5				
STATIC TEXT	5	220	164	ROUTE LENGTH (mi):	
NUMBER EDIT BOX	8	220	185		10
OK BUTTON	1	220	220	Enter	
CANCEL BUTTON	2	340	220	Cancel	

Figure 5-3 Dialog Box Definition Table

LINE HAUL INFO	
LINE HAUL MODE:	DECELERATION RATE (ft/sec ²):
Local Bus	13.12
BANDWIDTH (ft):	CRUISE VELOCITY (ft/sec):
1320	44
ROUTE HEADWAY (min):	AVERAGE TRIP LENGTH (mi):
30	4.5
ACCELERATION RATE (ft/sec ²):	ROUTE LENGTH (mi):
2.62	10
<input type="button" value="Enter"/> <input type="button" value="Cancel"/>	

Figure 5-4 First Operating Policy Dialog Box

The input boxes gather information for the first and fourth input forms listed in Table 5-8. In all cases and for the third form in a few places. As mentioned earlier, an input box in the *Backbone* routine asks for a file name for the summary statistics and one in each operating policy subroutine asks for a name for the results. None of them contain default values. *Backbone* also contains two input boxes that provide the user with an option and direct the program to respond according to the chosen option. The first, titled *New or Existing*, is in *Backbone* and asks the user if they "...wish to begin a new problem or examine/modify an existing one." The default is to begin a new problem. The latter option allows the user to work in the *Summcalc* subroutine. The *Operating Policy* input box asks users which of the operating policies they are going to utilize in their analysis and the response provides *Backbone* with its next instruction. Two is the default value because it applies to all four linehaul modes and the discrete system is the most common one in practice.

Furthermore, each of the operating policy subroutines contain the same input boxes and default values that serve to provide program user options. The first asks if the analysis is to be broken into a series of route segments with different characteristics. The boarding and deboarding demand, cruise velocity, segment length, and spacing increment are all parameters that can be influenced by the region (central business district, city center, or residential). The second asks if the dwell time will be entered or calculated with Poisson probability theory. Values entered into these boxes refer control to specific macro areas. For example, dialog boxes are presented to obtain variables for each segment and for the Poisson routine if these options are exercised. Different loops, moreover, are used to control repetition of calculations for

each spacing and replies to input queries for segmentation and dwell time tell the macro which of these loops to use.

Finally, input boxes request numerical values to be used in the calculations. They are similar for each operating policy subroutine although some of the default values change depending on the linehaul mode. Values for minimum spacing, maximum spacing, and spacing increment are requested via input boxes at the beginning of the *Typeone*, *Typetwo*, and *Typethree* subroutines. Default values are 400, 6000, and 400 feet (ft), respectively. Also, if the user chooses to enter an exact dwell or waiting time instead of the algorithm estimated values, they do so via an input box.

A majority of inputs are gained through dialog boxes. Each was created with a dialog box definition table (pictured in Figure 5-3) and saved in its own file. They are used for the second, third, and fourth input forms. Most of the dialog boxes are similar in format for each operating policy subroutine, but the default values and available access modes differ according to the linehaul mode. The two boxes for the fourth input type are identical, however, in each subroutine and ask the user how the algorithm is to interpret the meaning of the *W-value* and which formula to use to calculate the wait time. The response to the former directs control to a specific section in the macro to compute either user ingress and egress time for specific given distances or maximum acceptable distances.

The first dialog box that appears in each subroutine asks for the linehaul mode and route characteristics. Thus, it is unique to each operating policy since applicable modes for each are different. The route headway and length, vehicle acceleration and deceleration rates, cruise velocity, and average trip length are all obtained from this dialog box. Since there is a choice, the default values apply to the highlighted, or chosen, linehaul mode. Therefore, this dialog box and the walk rate dialog box are dynamic in that the default values change to reflect the highlighted option. The initial default values, thus, pertain to the default highlight. Once a different mode or rate is highlighted, the macro instantly redraws the dialog box with the new default values. The other dialog boxes for the second type of input, moreover, determine access modes the user wants to include in the analysis and, consequently, dictate which dialog box files the operating policy macro should open. Once these are opened, they ask for numerical values and modal split percents needed to calculate ingress and egress components of out-of-vehicle time. This information represents the majority of the inputs required for the algorithm and involves the most dialog boxes because of all the possible combinations of ingress and egress modes for each linehaul mode.

The other dialog boxes that obtain input values of the third type are for segmentation and Poisson dwell calculation options. The segment variables were previously mentioned and the macro displays a dialog box that includes them along with defaults. For each segment the default values are identical. Furthermore, the Poisson process requires values for boarding and alighting demand, start up times for these processes, and the probability of success. The middle and last parameters are constant for each segment, but the demand rates do vary. Thus, they appear in the dialog boxes for each segment if this

option is chosen and in a dialog box along with the last two if it is not. The format and default values of these boxes are the same in each operating policy subroutine.

5.1.3 Informative Messages

The Excel program provides an option of displaying messages in a status bar on the lower left hand screen corner. This algorithm exercises this option to inform the program users about what the program is doing as it runs the macro code. Table 5-10 provides an example of some of the user messages.

Table 5-10 Update Messages

**MESSAGE(TRUE,"Opening the
 necessary files.")**
**MESSAGE(TRUE,"Recording the walk access inputs to the summary
 statistics file.")**

MESSAGE(TRUE,"Computing the waiting time costs.")

Alert boxes are also displayed when necessary throughout the input process to convey various types of information. For instance, if an inappropriate input is entered in an input or dialog box, an alert box (Figure 5-5) explains the error. Other examples include entering negative numbers or text when a numerical value is required and vice versa. The macro redisplay the input or dialog box so the user can correct the error.

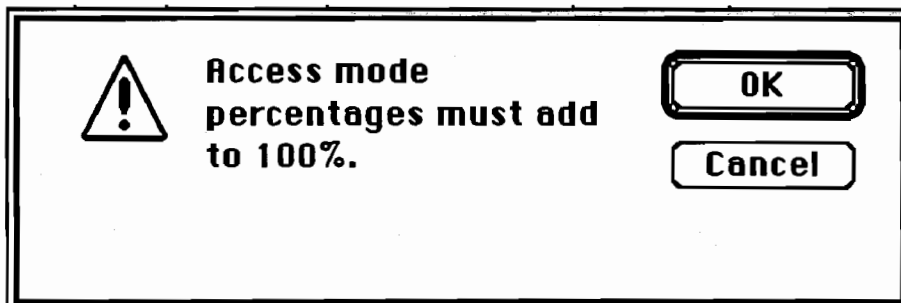


Figure 5-5 Alert Message Box

5.2 Program Sequence

The program follows a specific sequence through the main routine and the subroutines every time it is run. Figure 5-6 is a flowchart of this sequence and demonstrates how the program transfers control among the different macros. The tasks *Backbone* performs are in the ovals. The rectangles contain the options available to the user to tailor the analysis while the ones with the rounded ends display subroutine tasks.

When the program begins, *Backbone* asks the user if it is to start a new problem or allow the user to examine/modify an existing one. If the latter is chosen, the *Summcalc* macro file is opened and control transferred to it. The user then enters the name of the file that contains the inputs they wish to modify and *Summcalc* opens it while asking for names to use to save the modified input and results files. Once the user has changed the inputs on the summary sheet, the calculations to determine the travel time per spacing begin. The equations and the loop structure that guide the calculation sequence are the same here as they are in the other subroutines (subsequently discussed in detail). The only difference is that in-vehicle travel time is also computed within this macro instead of in the *Linehaul* subroutine. *Summcalc* retrieves all inputs the summary sheet and inserts them into the equations or uses them to direct control to a certain macro loop (fourth type of input). This subroutine compiles the modified results and saves them under the given name. This name should ideally be different from that for the original run so that comparisons can be made to determine the effect that the modified parameters have on user travel time.

On the other hand, if a new problem is to be initiated, *Backbone* opens a file to save a summary of all inputs while it asks for a name under which to save it. The user then chooses the operating policy to be utilized and that macro file is opened while this information is recorded on the summary sheet. *Backbone* receives the input as a number (1,2, or 3) and tests it with a series of **IF** statements to determine precisely which operating policy macro it needs to open. A **RUN** command then refers control to the subroutine. Each operating policy macro begins by opening all files that are common to every possible mode scenario for the policy type. The user then inputs a name to save the results, the minimum and maximum spacing, and the increment. After these values are recorded on the summary sheet, the linehaul information dialog box appears. If necessary, the macro redraws the dialog box to reflect the highlighted option and records the inputs on the summary sheet. The next dialog box asks the user how the algorithm should interpret the *W-value*. After this point, the input sequence is no longer identical for the three policies. In the *Typeone* and *Typethree* subroutines, the macro now requests the ingress, wait, segmentation, Poisson, and egress information via various input and dialog boxes. This order follows the order of the journey on public transportation. The wait time, segmentation, and Poisson input requests appear to the user after the linehaul mode and route information dialog box in the *Typetwo* subroutine. These parameters are common to all modes in the second operating policy, so they are gathered before control switches to the *Railtwo* subroutine, if necessary, since there is no need to build boxes for them into both routines.

Railtwo refers to *Typetwo* for these inputs. The ingress and egress dialog boxes are then presented in order in each of these two subroutines.

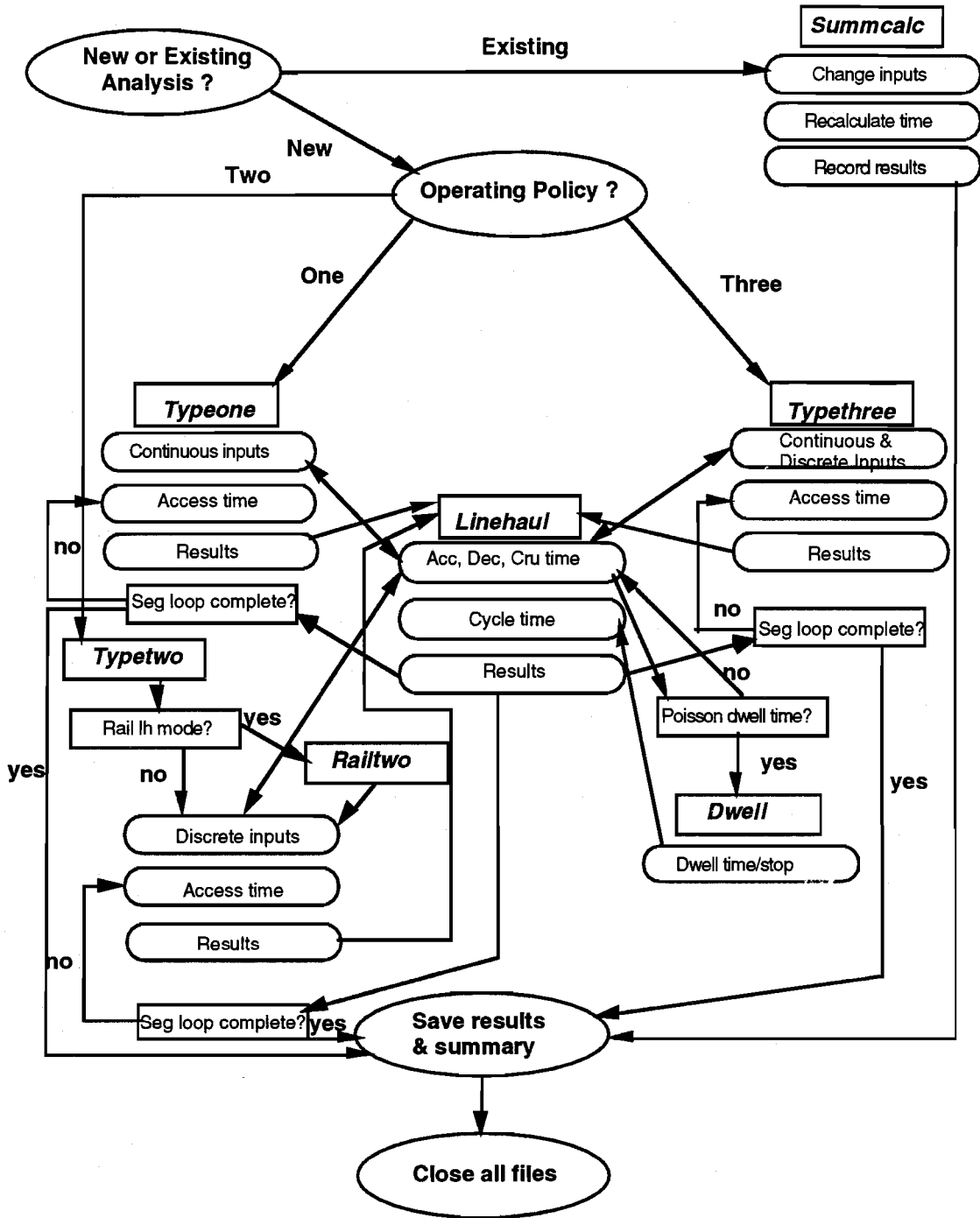


Figure 5-6 Program Control Flowchart

The next task for the first and third subroutines is opening the macro files that create the dialog boxes that are specific to the interpretation of the *W-value*. *Typethree* next presents an ingress mode dialog box. The name of the mode is received as a number by the macro. A series of IF statements uses this number as an argument to determine the chosen mode. These IF-END.IF loops contain instructions for displaying corresponding dialog boxes and recording inputs for the mode to the summary sheet. There is only one ingress mode in the first operating policy, so the macro displays the dialog boxes and records their inputs without one of these loops. The user next chooses the form of the wait time calculation and uses an input box to enter the exact waiting time if that is the chosen option.

The next input requests provide the user with options to dictate how the analysis is to be conducted. Two input boxes ask if the route will be segmented and if the dwell time will be calculated with Poisson probability theory. A series of four IF-END.IF loops then displays the required dialog boxes based on the two inputs. First, if there is no segmentation and the dwell time will be calculated, a dialog box for dwell time parameters appears and entered values are recorded on the summary sheet. Otherwise, the exact dwell time is determined with an input box. Segmentation, on the other hand, requires a number of dialog boxes. If the dwell time is to be calculated, the Poisson parameters appear in different dialog boxes. The ones that are constant are requested one time only and demand rates along with other parameters that vary across segments are requested in another dialog box. Code instructing the macro to display this box and record the values is in a loop that repeats one time for each segment. The user enters the desired number of segments through an input box. At the termination of the loop, an IF statement compares the sum of the segment lengths to the route length entered in the linehaul dialog box and commands the macro to repeat the input loop if they are not equal. Finally, if the user wants to input an exact dwell time, the dialog box with the segment variables does not include the boarding and alighting demand rates. The same looping process is repeated and an input box gains the dwell time once it finishes. All inputs are recorded to the summary sheet after specification through input and dialog boxes.

The final inputs are for the egress mode. A dialog box initially appears to determine the egress modes. The same one appears in both the *Typeone* and *Typethree* subroutines since the similar operating policy (continuous system) supports the same egress modes. The IF-END.IF loops perform the same function as for the ingress modes. For both sets of access inputs, an IF statement sums the percentages given to each mode if a combination is chosen and redisplay the dialog boxes if the total percent does not equal 100. The user sees the values just entered and can change them accordingly. Inputs are recorded to the summary sheet. After this, all inputs are known and users are not required to enter any more information.

The *Typetwo* operating policy subroutine is slightly different from the other two. After determining the meaning of the *W-value* for the analysis, the macro displays a dialog box for the user to indicate how the algorithm should calculate wait time. This subroutine utilizes the same code in the same

sequence as that previously described to obtain values for wait time, segmentation, and dwell time parameters. After they have been recorded to the summary sheet, the macro calculates wait time by taking the route headway value from the linehaul dialog box and inserting it into the appropriate formula if this is the specified wait time option. An IF statement determines if the linehaul mode is either light rail or rapid rail and, if so, opens the *Railtwo* macro file and sends control to it. The remaining ingress and egress inputs in that order are obtained in these two subroutines in the same manner as previously described for the first and third operating policy subroutines. The ingress and egress dialog boxes offer different mode choices depending on the linehaul mode. Figure 5-7 shows the dialog box that provides the egress mode choices for the bus rapid linehaul mode. The Local Bus/Walk mode is the default mode and it appears on the screen highlighted. Once the egress values have been entered and recorded on the summary sheet, the input process terminates.

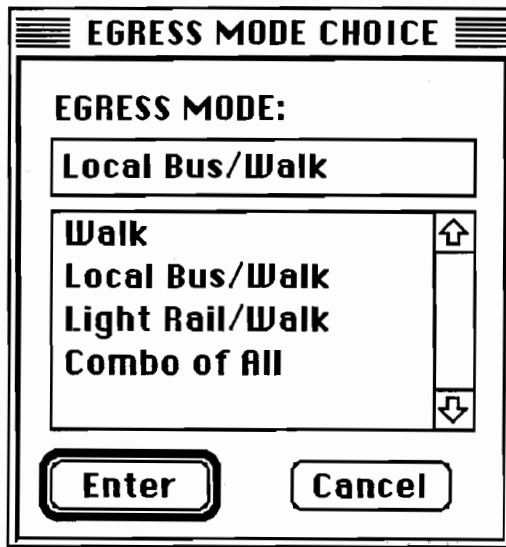


Figure 5-7 Egress Mode Dialog Box In Typetwo Subroutine

The subroutine macros employ a series of loops to repeat the travel time calculation for each station spacing. The input process is not included in a loop (with the exception of the segment variables) because the information is the same for every station spacing being tested. The only variable in the equations that changes is spacing (*S*) and it is specified with initial inputs for minimum, maximum, and increment distances. **WHILE-NEXT** loops control the iteration process for this task. The **WHILE** function sets a condition. In Table 5-11, *seg* is the counter (initially set to one (1)) and *B20* refers to the macro line that displays the input box asking for the number of segments. The counter is incremented by one (1) before the **NEXT** function is executed. The algorithm equations and formulas are in the lines between the two functions. The value of *seg* is compared to the number of route segments and if it is less

Table 5-11 WHILE - NEXT LOOP FOR SEGMENTATION

WHILE (OR (seg < B20, seg = B20)

SET.NAME ("seg", seg+1)

NEXT ()

than or equal, the condition is true continues running using values pertinent to the new segment number. As soon as the statement is false (*seg* is greater than the desired number of segments), the loop terminates and the macro jumps to the line of code beyond the **NEXT** function. The macro executes the entire algorithm for that segment in each pass through the loop. If there is no segmentation, the macro skips the formulas pertaining to the **WHILE-NEXT** loop. This, in effect, causes the macro to run through this loop only one time.

The segmentation loop has two sections. The first addresses the access (out-of-vehicle) time and the second the in-vehicle time. Even though the *Linehaul* macro is a separate subroutine, it is controlled by the segmentation loop because the operating policy subroutines call the *Linehaul* macro and refer control to it. Each section is contained within another **WHILE-NEXT** loop that governs the station spacing parameter. Table 5-12 is a picture of the code that formulates the loop. *Count* is the loop counter and is initially set to zero. *B10*, *B11*, and *B12* refer to input boxes created by these lines of code. They are the minimum, maximum, and increment, respectively, for the loop. The first iteration uses the minimum value for the spacing parameter (*S*). The increment is then added to the minimum spacing and this value is compared to the maximum spacing in the **WHILE** formula. If the condition is true, the loop repeats, the counter is incremented by one (1), and the spacing is tested again. The process terminates when the **WHILE** condition is false (the maximum allowable spacing has been exceeded) and the loop is skipped. The macro receives its next instruction from the formula on the line after the **NEXT** function. Each of these loops are also divided into two sections. The first calculates either the access or linehaul time and the second prints the results to the results worksheet.

Calculating the wait time is the first task of the access loop for the *Typeone* and *Typethree* subroutines. This has already been done in the *Typetwo* subroutine. An **IF** function then directs the macro to the set of equations pertinent to the meaning of the *W-value* since the ingress and egress equations are different for the two interpretations. The equations are built with instructions that tell the macro which dialog box has the value of the equation parameter to compute that time component time.

The dialog boxes are unique for each *W-value* interpretation and, thus, the references to them and the equations created from them are different.

Table 5-12 WHILE - NEXT LOOP FOR SPACING

WHILE (OR (B10 + count*B12 < B11, B10 + count*B12 = B12))

SET.NAME ("count", count+1)

NEXT()

The next **IF** statement within the access loop directs the macro where to record the calculated travel time components. One results sheet is formatted for one segment only for the case when the user does not choose the route segmentation option. The other sheet is formatted to record time components for each segment. An interactive graphing routine is built into these Excel worksheets and graphs travel time per station spacing as values are recorded from the subroutines.

After the results are recorded, the loop counter (*count*) is incremented by one and the **WHILE** condition (Table 5-12) is tested again. When it is false, the macro exits the access loop and is told to turn control over to *Linehaul* by the formula after the **NEXT** function.

The *Linehaul* subroutine is the second segmentation loop section. A set of two **IF** statements tell the macro which code section to use to calculate in-vehicle riding time and record results. Each of these four sections of equations is inside a **WHILE-NEXT** loop similar to the one in the segmentation loop access section. The **IF** statements determine if the analysis is segmented and if the dwell time is to be calculated with the Poisson procedure. The **IF** statement testing the former does not control the number of segments, rather it tells the macro which dialog boxes to refer to for the parameters in the equations. The other statement tells the macro how to obtain dwell time to add to the other three vehicle cycle time components. If the dwell time is given, the *Linehaul* macro retrieves the time from an input box displayed in the macro that called it. Otherwise, it transfers control to the *Dwell* macro. This subroutine pulls the values it needs for the Poisson procedure from the dialog boxes that are displayed in the operating policy macro that referred control to *Linehaul*. Once the dwell time per stop is determined, control returns to the *Linehaul* macro and the in-vehicle riding time is calculated. This travel time component is sent to the results worksheet with the same routine that was used for the access time components. The loop counter (*count*) is incremented by one and the **WHILE** condition tested again. The macro runs through the same

set of equations until the **WHILE** condition becomes false. Once the loop is exited, the ensuing formula reverts control of the program back to the operating policy subroutine that originally transferred control to the *Linehaul* subroutine.

The algorithm has been completed for one segment once *Linehaul* returns control. The loop counter (*seg*) is incremented by one and the first **WHILE** condition (Table 5-11) is tested again. If it is true, the entire process repeats. The equations that contain the parameters that vary by segment tell the macro where to get values for a particular segment under analysis. If the route is not segmented, a formula instructs the macro to skip the formulas that increment the segment counter and direct the program back up to the **WHILE** statement. The final line of code in the operating policy subroutines returns program control program to the *Backbone* routine.

When the subroutines finish calculating user travel time and assembling results, they return control to the main routine and *Backbone* saves results with names input in the operating policy subroutines. Again, a series of **IF** statements is needed to determine which of two possible results sheets is to be saved. One format accommodates results of an analysis that includes segmentation and the other for an analysis that does not. *Backbone* refers to the word returned in the input box (yes or no) that inquires about breaking the route into segments for the argument in the **IF** statement. The final task for this macro is to close all open files without allowing any protected files to be changed.

5.3 Algorithm Results

The results file contains both a table and a graph of user travel time per station spacing. Figure 5-8 is an example of an analysis without segmentation for an average trip length. The table lists spacing and each travel time component for an average trip length (four and one-half (4.5) miles in this analysis) along with total access and travel times for that spacing. Interstation spacing distances are in feet and time components in minutes. If the route is segmented, the first column contains the segment number and the totals columns refer to travel time for the segment only. There is one summary table and graph for each segment. The time to travel for an average trip length is not included in the segmentation analysis since most passengers will ride through parts of multiple segments. The shortest overall travel time equates to lowest user cost and corresponding spacing is the optimal distance between transit terminals. In this analysis, a station spacing of 2400 feet (ft) yields the shortest travel time of 55.43 minutes (min) for an average trip length. The figure is a graph of information in this table with station spacing in feet on the horizontal axis and user travel time in minutes on the vertical axis. Again, there is one for each segment if necessary. Total access time (square data markers) and linehaul travel time (triangular data markers) are both graphed versus station spacing. The third curve (diamond data markers) represents total overall travel time per station spacing and is the sum of access and linehaul curves. The minima of this curve is

SPACING	INGRESS TIME	WAIT TIME	AVG LH TIME	EGRESS TIME	ACCESS TIME	TOTAL TIME
400	4.94	4.47	51.34	24.66	34.07	85.41
800	5.11	4.47	35.62	24.92	34.50	70.12
1200	5.37	4.47	26.44	25.33	35.17	61.61
1600	5.72	4.47	21.85	25.84	36.03	57.88
2000	6.13	4.47	19.10	26.44	37.04	56.14
2400	6.61	4.47	17.26	27.09	38.17	55.43
2800	7.13	4.47	16.34	27.78	39.38	55.72
3200	7.68	4.47	15.43	28.51	40.66	56.09
3600	8.27	4.47	14.51	29.26	42.00	56.51
4000	8.88	4.47	13.59	30.03	43.38	56.97

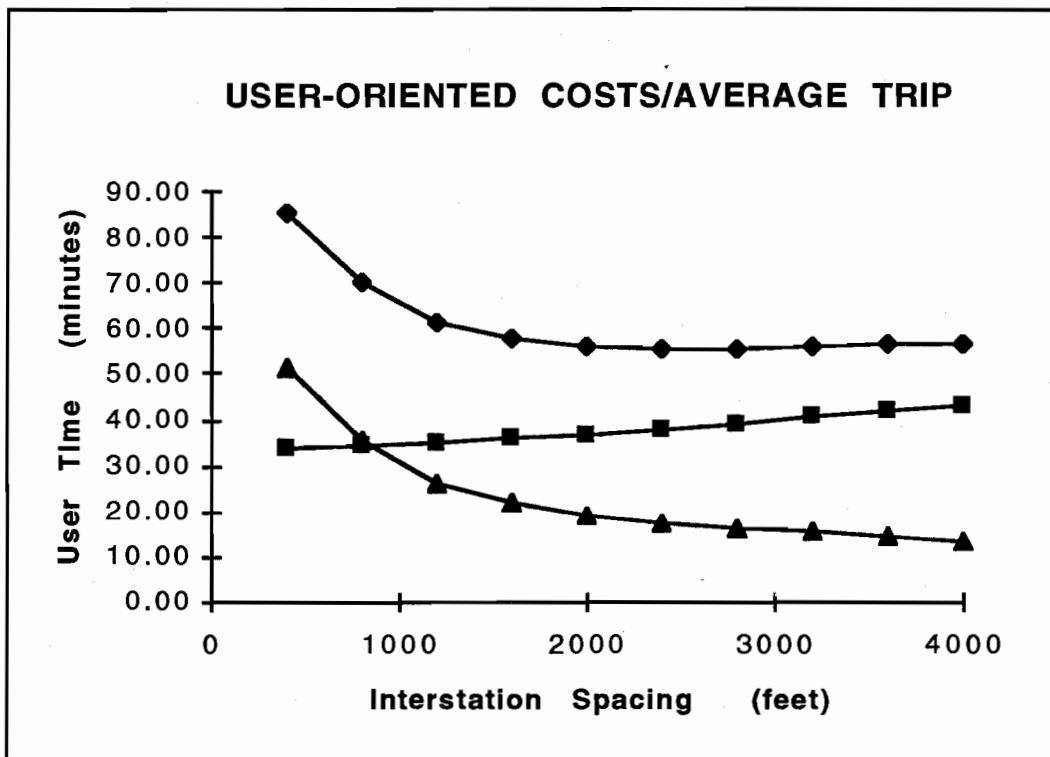


Figure 5-8 User Travel Time Costs For An Average Trip Length

the shortest travel time and represents a compromise solution between the two user time components for optimal interstation spacing. This distance will provide users with the lowest possible cost for a system designed according to their preferences and needs.

Figure 5-9 presents results of the same analysis for the entire route length of ten (10) miles (mi). Access time is again represented by square data markers and linehaul time is indicated by the triangular data markers. The optimal interstation spacing according to the total sum curve (diamond data markers) is 3600 feet (ft).

SPACING	ACCESS TIME	WAIT TIME	ROUTE LH TIME	EGRESS TIME	ACCESS TIME	TOTAL TIME
400	4.94	4.47	114.68	24.66	34.07	148.76
800	5.11	4.47	80.58	24.92	34.50	115.08
1200	5.37	4.47	60.39	25.33	35.17	95.56
1600	5.72	4.47	50.29	25.84	36.03	86.32
2000	6.13	4.47	43.87	26.44	37.04	80.91
2400	6.61	4.47	40.19	27.09	38.17	78.36
2800	7.13	4.47	36.52	27.78	39.38	75.90
3200	7.68	4.47	34.69	28.51	40.66	75.35
3600	8.27	4.47	32.85	29.26	42.00	74.85
4000	8.88	4.47	31.93	30.03	43.38	75.31

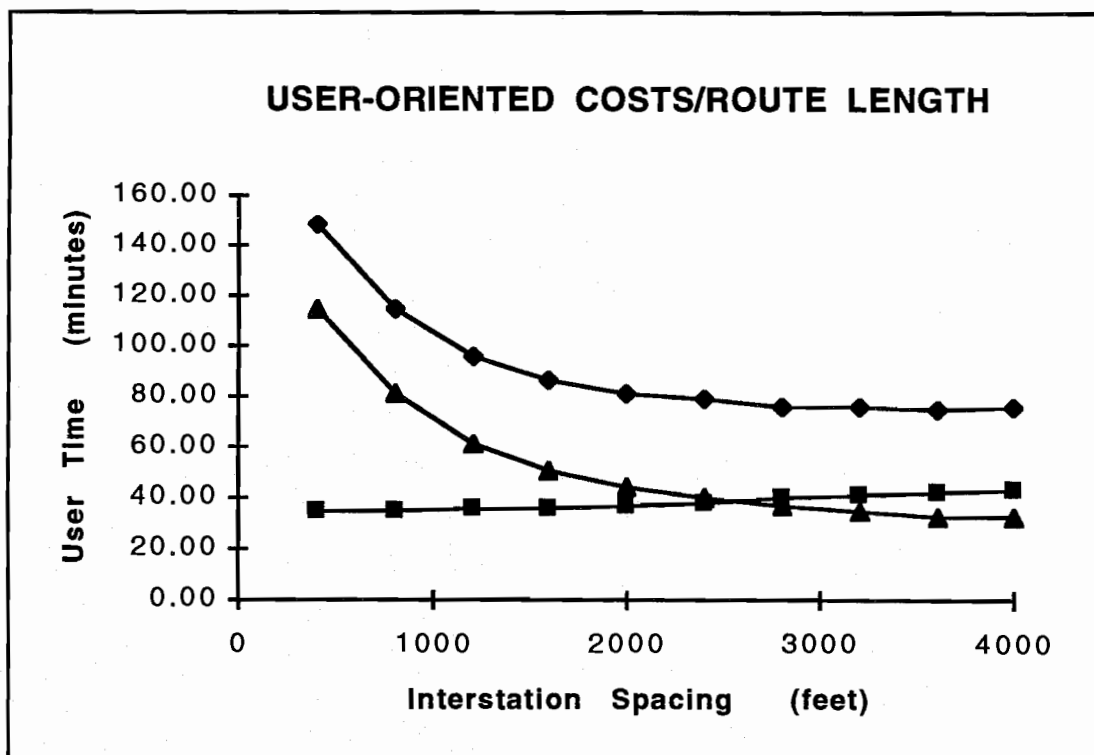


Figure 5-9 User Travel Time Costs For The Route Length

This large distance shows that the access distance that the users are willing to travel is very short compared to the route length and, hence, in-vehicle travel time over the whole route length is extremely large compared to access time. The summary table also shows that 3600 feet (ft) is optimal since it provides the shortest time of 74.85 minutes (min). While an analysis of the time to cover the route is useful to study the operator costs, it should be noted that few passengers will ever need to travel the entire route length and the expected average trip length results should be weighed more heavily.

The route length and average trip length are input in the linehaul dialog box. This scenario depicts a discrete system for which local bus is the linehaul mode. Walking is the ingress mode and a combination of walking and transferring to another local bus is the egress mode.

5.4 Summary

This customized Excel program implements optimal station spacing analysis algorithm quickly and accurately. More circumstances for user preferences can be tested with it than is feasible if the analysis is done by hand. The possibility of error is also reduced through this computerized calculation procedure.

CHAPTER 6

CONCLUSION

A public transportation system is an important element of the urban environment since it contributes to social and environmental welfare. It provides mobility to those who cannot drive or have access to an automobile. Transit also provides an alternative mode to driving for those who have a choice. When people take advantage of the system, it helps alleviate congestion, pollution, and parking shortage problems common to urban areas. In order to be effective at accomplishing these objectives, however, the system must be appealing enough to attract riders. Needs and preferences of potential users must therefore be addressed when designing a transit system.

Terminal location and spacing is one of the most crucial features of the system design because it has a profound impact upon user travel time as well as upon overall system performance. This travel time is one of the most effective means of influencing user travel modes. Therefore, choices regarding terminal location and frequency must consider the component parts of user travel time and attempt to minimize them. This approach also benefits the system operator because it will result in enhanced ridership.

6.1 Concepts

Terminal spacing has a direct effect on almost all components of out-of-vehicle and in-vehicle time. The literature review leads to the hypothesis that the two vary inversely when the distance between terminals increases or decreases. For example, linehaul travel time, or time spent riding in the vehicle, increases as the interstation spacing decreases. Transit vehicles travel through more stopping cycles and encounter more stop-related delays with this spacing policy. Access time, on the other hand, increases as the interstation spacing increases because the same volume of users has to travel farther to reach fewer terminals. Thus, the calculation of these times necessarily includes terminal spacing. This research built an algorithm from these concepts to determine optimal interstation spacing based on user travel time. It employs values from the literature to represent variables in equations that estimate access and linehaul travel times. The algorithm calculates this time for several possible station spacings. The shortest of these equates to the lowest user cost and yields the optimal interstation spacing. Results indicate that the hypothesized relationship is true and holds for any linehaul mode and combination of access modes. This research project also computerized the algorithm in a version of the Microsoft Excel™ program. The program enables creation of a wide variety of scenarios for different operating policies, access and linehaul mode combinations, and user preferences. This feature makes the program applicable in any area regardless of geographical location or population size.

6.2 Applications

The program is a practical tool for to optimizing transit terminal spacing. Its friendly interface guides users through an input process that easily enables them to tailor the analysis to their unique situation. As such, the results can aid in the development and enhancement of public transportation policies. They are capable of producing a foundation for the design of new systems that will attract riders and support continued growth while still controlling costs. Examining different operating policy options is additionally useful in determining which one is most feasible for particular user preferences. Moreover, the program is useful in analyzing existing systems and will indicate any modifications that will reduce costs and enhance ridership. Although the final system policy is subjected to a multitude of real world constraints, the program results can serve as an objective basis for decisions regarding terminal spacing design.

6.3 Recommendations

In the future, the computerized algorithm would benefit from a more in-depth study of all default values. This may be achieved by collecting primary data on each travel time component that describes user behavior and assembling a large database. From it, models for some of the time components could be constructed. Furthermore, values taken from resulting distributions, can better represent components. This effort should gather data for each of the four linehaul modes and encompass several transit systems and cities.

REFERENCES

1. Abkowitz, M.D. and Israel Engelstein. "Factors Affecting Running Times on Transit Routes." Transportation Research A: General 17A.2 (1983): 107-113.
2. Abkowitz, Mark and John Tozzi. "Transit Route Characteristics and Headway-Based Reliability Control." Transportation Research Record 1078 (1986): 11-16.
3. Demetsky, Michael and Jerry Korf. "Analysis of Rapid Transit Access Mode Choice." Transportation Research Record 817 (1981): 29-35.
4. Demetsky, Michael and Michael Perfater. "Assessment of Pedestrian Attitudes and Behavior in Suburban Environments." Transportation Research Record 540 (1975): 48-49.
5. Feder, R.C. "Effect of Bus Stop Spacing and Location on Travel Time." Master's thesis, Carnegie Mellon University, 1978.
6. Ghoneim, N.S. and S.C. Wirasinghe. "Spacing Bus Stops for Many to Many Travel Demand." Transportation Science 15.3 (1981): 210-221.
7. Guenther, R.P. and Kumares Sinha. "Modeling Bus Delays Due to Passenger Boardings and Alightings." Transportation Research Record 915 (1983): 7-13.
8. Harvey, J. T. "Some Aspects of Rapid Transit Station Design." Paper. ASCE Transportation Facilities Workshop: Passenger, Freight, and Parking. New York, 22-24 May, 1974.
9. Hoel, Lester. "Pedestrian Travel Rates in Central Business Districts." Traffic Engineering January 1968: 10-13.
10. Homburger, Wolfgang., ed. Transportation and Traffic Engineering Handbook. 2nd ed. New Jersey: Prentice-Hall, Inc., 1982.
11. Kraft, W.H. and T.F. Bergen. "Evaluation of Passenger Service Times for Street Transit Systems." Transportation Research Record 505 (1974): 13-20.

12. Kuah, Geok and Jossef Perl. "Optimization of Feeder Bus Routes and Bus-Stop Spacing." Journal of Transportation Engineering 114 (1988): 341-354.
13. Lesley, L.T.S. "Optimum Bus Stop Spacing, Part One." Traffic Engineering and Control October 1976: 399-401.
14. Levinson, Herbert. "Analyzing Transit Travel Time Performance." Transportation Research Record 915 (1983): 1-6.
15. Machemehl, Randy. Lecture. Public Transportation Engineering Class. University of Texas, 24 March 1992.
16. Middleton, William. "LRT Helps Shape A City." Railway Age February 1990: LR2-LR11.
17. Nungesser, Lisa and Nancy Ledbetter. Procedures For Estimating Park and Ride Demand in Large Texas Cities. Texas: Texas Transportation Institute #2-10-84-339-7, 1987.
18. Petersen, Stephen. "Walking Distances To Bus Stops in Washington, D.C. Residential Areas." Traffic Engineering December 1968: 28-34.
19. Polus, Abishai and Daniel Shefer. "Evaluation of a Public Transportation Level of Service Concept." Journal of Advanced Transportation 18.2 (1984): 135-144.
20. Pushkarev, Boris and Jeffrey Zupan. Public Transportation and Land Use Policy. Indiana: Indiana University Press, 1977.
21. Seneviratne, P.N. "Acceptable Walking Distances in Central Areas." Journal of Transportation Engineering 915 (1983): 7-13.
22. Seneviratne, P.N. and Chu Chai Loo. "Bus Journey Times in Medium Sized Urban Areas." Journal of Advanced Transportation Engineering 20.3 (1986): 259-274.
23. Shortreed, John. Urban Bus Transit: A Planning Guide. Ontario: University of Waterloo, 1974.

24. Stringham, M.G.P. "Travel Behavior Associated with Land Uses Adjacent to Rapid Transit Stations." ITE Journal April 1982: 16-18.
25. Vuchic, Vucan. "Interstation Spacings for Linehaul Passenger Transportation." Dissertation. University of California at Berkeley, 1966.
26. Vuchic, Vucan. "Rapid Transit Interstation Spacings for Maximum Number of Passengers." Transportation Science 3.3 (1968): 214-232.
27. Vuchic, Vucan. Urban Public Transportation. New Jersey: Prentice-Hall, Inc., 1981.
28. Vuchic, Vucan and G.F. Newell. "Rapid Transit Interstop Spacings for Minimum Travel Time." Transportation Science 11.4 (1968): 359-374.
29. Winick, Robert and Steven Smith. "Short Term Impacts of A Suburban Rail Rapid Transit Station: Study Results for Silver Spring, Maryland." Transportation Research Record 760 (1980): 1-7.
30. Wirasinghe, S.C. "Nearly Optimal Parameters for a Rail Feeder Bus System on a Rectangular Grid." Transportation Research A: General 14A.1 (1980): 33-40.

APPENDIX A

**THE OPTIMAL TRANSIT
TERMINAL SPACING MACRO CODE**

BACKBONE MAIN ROUTINE

Backbone Main Routine

<i>name</i>	Backbone =MESSAGE(FALSE) =ECHO(FALSE)
<i>prompt for new or old problem</i>	=INPUT("Do you wish to begin a new problem (new) or examine/modify an existing one (existing)?",2,"NEW OR EXISTING", "new")
<i>if new problem</i>	=IF(B4="new") =MESSAGE(TRUE,"Opening the necessary files.") =OPEN("JAD HD:REAL DEAL:Summary",TRUE)
<i>name to save summary sheet</i>	=INPUT("Enter a name under which to save the summary statistics.",2,"SUMMARY STATISTICS")
<i>goto first input box if existing problem</i>	=GOTO(B17) =ELSE() =MESSAGE(TRUE,"Opening the necessary files.")
<i>goto macro that will allow for changes and recalculations</i>	=OPEN("JAD HD:REAL DEAL:Summcalc",TRUE) =RUN(!B2) =END.IF()
<i>go to end to close all files,exit</i>	=GOTO(B81) =MESSAGE(FALSE)
<i>input operating policy type</i>	=INPUT("Enter policy type number:",1,"OPERATING POLICY",2) =IF(B17>3) =ALERT("Selection must be a number between 1 and 3.",2) =GOTO(B17) =END.IF() =COPY(B17) =ACTIVATE("Summary") =SELECT('JAD HD:REAL DEAL:Summary'!B2)
<i>print operating policy number</i>	=PASTE.SPECIAL(3,1) =ACTIVATE("Backbone")
<i>direct program to proper turn off screen updating</i>	=IF(B17=1) =ECHO(FALSE) =MESSAGE(TRUE,"Opening the necessary files.")
<i>open Typeone2 to read only subroutine for access/egress and wait times</i>	=OPEN("JAD HD:REAL DEAL:Typeone",TRUE) =RUN('JAD HD:REAL DEAL:Typeone'!B2) =END.IF() =IF(B17=2)
<i>turn off screen updating open Typetwo to read only</i>	=ECHO(FALSE) =OPEN("JAD HD:REAL DEAL:Typetwo",TRUE) =RUN('JAD HD:REAL DEAL:Typetwo'!B2) =END.IF() =IF(B17=3)
<i>turn off screen updating open Typethree to read only</i>	=ECHO(FALSE) =OPEN("JAD HD:REAL DEAL:Typethree",TRUE) =RUN('JAD HD:REAL DEAL:Typethree'!B2)

	=END.IF()
	=ECHO(FALSE)
	=MESSAGE(TRUE,"Saving the summary statistics and results.")
	=ACTIVATE("Summary")
	=B8
<i>save summary sheet</i>	=SAVE.AS(B46)
	=IF(B17=1)
<i>no segmentation</i>	=IF('JAD HD:REAL DEAL:Typeone'!B194="no")
	=ACTIVATE("WORKSHEET")
	='JAD HD:REAL DEAL:Typeone'!B8
<i>save results</i>	=SAVE.AS(B51)
<i>with segmentation</i>	=ELSE()
	=ACTIVATE("WORKSHEETS")
	='JAD HD:REAL DEAL:Typeone'!B8
<i>save results</i>	=SAVE.AS(B55)
<i>end segmentation loop</i>	=END.IF()
<i>end typeone loop</i>	=END.IF()
<i>Typetwo loop</i>	=IF(B17=2)
<i>no segmentation</i>	=IF('JAD HD:REAL DEAL:Typetwo'!B68="no")
	=ACTIVATE("WORKSHEET")
<i>all modes</i>	='JAD HD:REAL DEAL:Typetwo'!B6
<i>save results</i>	=SAVE.AS(B62)
<i>segmentation</i>	=ELSE()
	=ACTIVATE("WORKSHEETS")
<i>all modes</i>	='JAD HD:REAL DEAL:Typetwo'!B6
<i>save results</i>	=SAVE.AS(B66)
<i>end segmentation loop</i>	=END.IF()
<i>end Typetwo loop</i>	=END.IF()
<i>begin typethree loop</i>	=IF(B17=3)
<i>no segmentation</i>	=IF('JAD HD:REAL DEAL:Typethree'!B157="no")
	=ACTIVATE("WORKSHEET")
	='JAD HD:REAL DEAL:Typethree'!B6
<i>save results</i>	=SAVE.AS(B73)
<i>with segmentation</i>	=ELSE()
	=ACTIVATE("WORKSHEETS")
	='JAD HD:REAL DEAL:Typethree'!B6
<i>save results</i>	=SAVE.AS(B77)
<i>end segmentation loop</i>	=END.IF()
<i>end typethree loop</i>	=END.IF()
<i>save summcalc results</i>	=IF(B4="existing")
	=ACTIVATE("Summcalc")
	=ACTIVATE(!B3)
<i>no segmentation</i>	=IF(!D2="")
	=ACTIVATE("Worksheet")
	='JAD HD:REAL DEAL:Summcalc'!B8
	=SAVE.AS(B86)

```
end segmentation loop
=ELSE()
=ACTIVATE("WorksheetS")
='JAD HD:REAL DEAL:Summcalc'!B8
=SAVE.AS(B90)
=END.IF()
=ACTIVATE("Summcalc")
=ACTIVATE(!B3)
=SAVE.AS('JAD HD:REAL DEAL:Summcalc'!B7)
end summcalc loop
=END.IF()
=MESSAGE(FALSE)
=ERROR(FALSE)
prevent windows from showing
=CLOSE.ALL()
close all linked macros
=RETURN()
```


Summcalc Subroutine

<i>macro to calculate from summ</i>	Summcalc
	=MESSAGE(FALSE)
<i>name of file</i>	=INPUT("Enter the name of the file under which the summary statistics are saved.",2,"SUMMARY STATISTICS")
	=B3
<i>status bar message</i>	=MESSAGE(TRUE,"Opening the file with the summary statistics.")
<i>open the file</i>	=OPEN(B4)
<i>name of file</i>	=INPUT("Enter the name of the file under which to save the summary statistics for this run.",2,"SUMMARY STATISTICS")
<i>name of file</i>	=INPUT("Enter the name of the file under which to save the results of this run.",2,"SUMMARY STATISTICS")
<i>status bar message</i>	=MESSAGE(TRUE,"Opening the results files.")
<i>open results WORKSHEET</i>	=OPEN("JAD HD:REAL DEAL:WORKSHEET",0,TRUE)
<i>open results WORKSHEETS</i>	=OPEN("JAD HD:REAL DEAL:WORKSHEETS",0,TRUE)
<i>status bar message</i>	=MESSAGE(TRUE,"Initializing the loop counters.")
<i>increment column print counter</i>	=SET.NAME("answer",0)
<i>increment segment counter</i>	=SET.NAME("seg",1)
	=MESSAGE(FALSE)
<i>skip counter stuff for solo</i>	=IF(!D2="")
	=GOTO(B27)
	=END.IF()
	=WHILE(OR(seg<!D2,seg=!D2))
	=answer
	=seg
	=COPY(B21)
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$A\$3,0,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>initialize spacing counter to zero</i>	=SET.NAME("count",0)
	=ACTIVATE(B4)
	=OFFSET(!K\$12,seg,0)
<i>test conditions to continue loop</i>	=WHILE(OR(!E4+count*B29<!G4,!E4+count*B29=!G4))
	=ACTIVATE(B4)
<i>display spacing</i>	=!E4+(B29*count)
<i>status bar message</i>	=MESSAGE(TRUE,"Computing the access and egress costs.")
<i>operating policy Typeone access</i>	=IF(!B2=1)
<i>willing to travel w value</i>	=IF(!B5="Distance willing to travel")
<i>access time in minutes</i>	=((((!F6^2)+((!D3/2)^2)^0.5)/(!D6*60))
<i>actual distance</i>	=ELSE()
<i>access time in minutes</i>	=(!H6/(!D6*60))
<i>end w value loop</i>	=END.IF()
<i>end Typeone access loop</i>	=END.IF()

Typeone or three policy egress	=IF(OR(!B2=1,!B2=3))
walk egress	=IF(OR(!B20="Walk",!B20="Walk (Adults)",!B20="Walk (Elderly)",!B20="Walk (Children)",!B20="Walk (CBD)"))
if willing to travel w value	=IF(!B5="Distance willing to travel")
egress time in minutes	=((((!(D20^2)+((!D3/2)^2))^0.5)/(!D20*60))
actual distance	=ELSE()
walk egress time in minutes	=(!H20/(!D20*60))
end w value loop	=END.IF()
end walk egress loop	=END.IF()
bus egress time in minutes	=IF(!B20="Local Bus/Walk")
willing to travel w value	=IF(!B5="Distance willing to travel")
	=(!F\$21/(!D\$21/60))+(!J\$21^0.5)+(((!P\$21^2)+((!D3/2)^2))^0.5)/(!N\$21*60))
actual distance	=ELSE()
bus egress time in minutes	=(!H\$21/(!D\$21/60))+(!J\$21^0.5)+(!R\$21/(!N\$21*60))
end w value loop	=END.IF()
end bus egress loop	=END.IF()
combo egress time in minutes	=IF(!B20="Combo of Both")
willing to travel w value	=IF(!B5="Distance willing to travel")
egress time (walk)	=(!L\$20/100)*((((!F\$20^2)+((!D3/2)^2))^0.5)/(!D\$20*60)))
egress time (bus)	=(!L\$21/100)*((!F\$21/(!D\$21/60))+(!J\$21^0.5)+(((!P\$21^2)+((!D3/2)^2))^0.5)/(!N\$21*60)))
egress time (combo)	=B58+B59
actual distance	=ELSE()
egress time (walk)	=(!L\$20/100)*((!H\$20/(!D\$20*60)))
egress time (bus)	=(!L\$21/100)*((!H\$21/(!D\$21/60))+(!J\$21^0.5)+(!R\$21/(!N\$21*60)))
egress time (combo)	=B62+B63
end w value loop	=END.IF()
end combo egress loop	=END.IF()
end Typeone or three egress loop	=END.IF()
Typetwo access and egress loop	=IF(!B2=2)
walk access	=IF(!B6="Walk")
if willing to travel w value	=IF(!B5="Distance willing to travel")
access time in minutes	=((((!F\$6^2)+((B\$32/2)^2))^0.5)/(!D\$6*60))
actual distance	=ELSE()
walk access time in minutes	=(!H\$6/(!D\$6*60))
end w value loop	=END.IF()
end walk access loop	=END.IF()
auto access	=IF(!B6="Automobile")
auto access time in minutes	=!D7+!J7
end auto access loop	=END.IF()
bus access time in minutes	=IF(!B6="Local Bus/Walk")
willing to travel w value	=IF(!B5="Distance willing to travel")
access time in minutes	=((((!F\$8^2)+(((B\$32/2)/5280)^2))^0.5)/(!D\$8/60))+((((!P\$8^2)+((B\$32/8)^2))^0.5)/(!N\$8*60))+(!J\$8^0.5)
actual distance	=ELSE()

<i>bus access time in minutes</i>	$=((!H\$8/(!D\$8/60))+(!J\$8^0.5))$
<i>end w value loop</i>	$=END.IF()$
<i>end bus access loop</i>	$=END.IF()$
<i>rail access time in minutes</i>	$=IF(!B6="Light Rail/Walk")$
<i>willing to travel w value</i>	$=IF(!B12="Distance willing to travel")$
<i>rail ride/wait</i>	$=((!F9/(!D9/60))+(!J9^0.5))$
<i>walk to rail</i>	$=((!T9/100)*(((!P9^2)+((B\$32/8)^2)^0.5)/(!N8*60)))$
<i>drive to rail</i>	$=(!V9/100)*(!D7+!J7)$
<i>bus to rail</i>	$=(((X9/100)*(((P9^2)+((B\$32/8)/5280)^2)^0.5)/(!D8/60))+(((P8^2)+(B\$32/16)^2)^0.5)/(!N9*60))+(!J8^0.5)$
<i>total time</i>	$=B88+B89+B90+B91$
<i>actual distance</i>	$=ELSE()$
<i>rail access time in minutes</i>	$=((!T9/100)*(!R9/(!N9*60)))+((!V9/100)*(!D7+!J7))+((!X9/100)*(!F8/!D8/60))+(!J9^0.5)+(!H9/(!D9/60))$
<i>end w value loop</i>	$=END.IF()$
<i>end rail access loop</i>	$=END.IF()$
<i>combo access time in minutes</i>	$=IF(!B6="Combo of All")$
<i>willing to travel w value</i>	$=IF(!B5="Distance willing to travel")$
<i>combo access time (walk)</i>	$=(!L\$6/100)*(((!F\$6^2)+((B\$32/2)^2)^0.5)/(!D\$6*60))$
<i>combo access time (auto)</i>	$=(!L\$7/100)*(!D\$7+!J\$7)$
<i>combo access time (bus)</i>	$=(!L\$8/100)*(((!F\$8^2)+((B\$32/2)/5280)^2)^0.5)/(!D\$8/60))+((!R\$8^2)+((B\$32/8)^2)^0.5)/(!N\$8*60))+(!J\$8^0.5)$
<i>combo access time (rail ride/wait)</i>	$=((!F9/(!D9/60))+(!J9^0.5))$
<i>walk to rail</i>	$=((!T9/100)*(((!R9^2)+((B\$32/8)^2)^0.5)/(!N9*60)))$
<i>drive to rail</i>	$=(!V9/100)*(!D7+!J7)$
<i>bus to rail</i>	$=((!X9/100)*(((F9/5280)^2+(B\$32/8)^2)^0.5)/(!D6/60))+(((P8^2)+(B\$32/16)^2)^0.5)/(!N8*60))+(!J8^0.5))$
<i>total rail time</i>	$=(!L9/100)*(B102+B103+B104+B105)$
<i>total combo time</i>	$=B99+B100+B101+B106$
<i>actual distance</i>	$=ELSE()$
<i>combo access time (walk)</i>	$=(!L\$6/100)*((!H\$6^2)/(!D\$20*60))$
<i>combo access time (auto)</i>	$=(!L\$7/100)*(!D\$7+!J\$7)$
<i>combo access time (bus)</i>	$=(!L\$8/100)*((!H\$8/(!D\$8/60))+(!J\$8^0.5)+(!R\$8/(!N\$8*60)))$
<i>combo access time (rail)</i>	$=(!L9/100)*(((T9/100)*(!R9/(!N9*60)))+((V9/100)*(!D7+!J7))+((X9/100)*(!H8/(!D7/60))+(!J8^0.5)))+(!J9^0.5)+(!H9/(!D9*60)))$
<i>combo access time (total)</i>	$=B109+B110+B111+B112$
<i>end w value loop</i>	$=END.IF()$
<i>end combo access loop</i>	$=END.IF()$
<i>walk egress loop</i>	$=IF(!B20="Walk")$
<i>if willing to travel w value</i>	$=IF(!B5="Distance willing to travel")$
<i>egress time in minutes</i>	$=(((F\$20^2)+((B\$32/2)^2)^0.5)/(!D\$20*60))$
<i>actual distance</i>	$=ELSE()$
<i>walk egress time in minutes</i>	$=(!H\$20/(!D\$20*60))$
<i>end w value loop</i>	$=END.IF()$
<i>end walk egress loop</i>	$=END.IF()$
<i>bus egress time in minutes</i>	$=IF(!B20="Local Bus/Walk")$
<i>willing to travel w value</i>	$=IF(!B5="Distance willing to travel")$

<i>bus egress time in minutes</i>	$=(!F\$21/(!D\$21/60))+(!J\$21^0.5)+(((!P\$21^2)+((B\$32/2)^2))^0.5)/(!N\$21*60))$
<i>actual distance</i>	$=ELSE()$
<i>bus egress time in minutes</i>	$=(!H\$21/(!D\$21/60))+(!J\$21^0.5)+(!R\$21/(!N\$21*60))$
<i>end w value loop</i>	$=END.IF()$
<i>end bus egress loop</i>	$=END.IF()$
<i>rail egress time in minutes</i>	$=IF(!B20="Light Rail/Walk")$
<i>willing to travel w value</i>	$=IF(!B5="Distance willing to travel")$
<i>rail egress time in minutes</i>	$=(!F22/(!D22/60))+(!J22^0.5)+(((!R22^2)+((B\$32/2)^2))^0.5)/(!N22*60))$
<i>actual distance</i>	$=ELSE()$
<i>rail egress time in minutes</i>	$=(!H22/(!D22/60))+(!J22^0.5)+(!R22/(!N22*60))$
<i>end w value loop</i>	$=END.IF()$
<i>end rail egress loop</i>	$=END.IF()$
<i>combo egress time in minutes</i>	$=IF(!B20="Combo of All")$
<i>willing to travel w value</i>	$=IF(!B5="Distance willing to travel")$
<i>combo egress time (walk)</i>	$=(!L\$20/100)*(((!F\$20^2)+((B\$32/2)^2))^0.5)/(!D\$20*60))$
<i>combo egress time (bus)</i>	$=(!L\$21/100)*(((!F\$21/(!D\$21/60))+(!J\$21^0.5)+(((!P\$21^2)+((B\$32/2)^2))^0.5)/(!N\$21*60))$
<i>combo egress time (rail)</i>	$=(!L22/100)*(((!F22/(!D22/60))+(!J22^0.5)+(((!P22^2)+((B\$32/8)^2))^0.5)/(!N22*60))$
<i>combo egress time (total)</i>	$=B139+B140+B141$
<i>actual distance</i>	$=ELSE()$
<i>combo egress time (walk)</i>	$=(!L\$20/100)*(!H\$20/(!D\$20*60))$
<i>combo egress time (bus)</i>	$=(!L\$21/100)*(!H\$21/(!D\$21/60))+(!J\$21^0.5)+(!R\$21/(!N\$21*60))$
<i>combo egress time (rail)</i>	$=(!L22/100)*(!H22/(!D22/60))+(!J22^0.5)+(!R22/(!N22*60))$
<i>combo egress time (total)</i>	$=B144+B145+B146$
<i>end w value loop</i>	$=END.IF()$
<i>end combo egress loop</i>	$=END.IF()$
<i>end Typetwo access/egress loop</i>	$=END.IF()$
<i>Typetwo access loop</i>	$=IF(!B2=3)$
<i>walk access</i>	$=IF(!B6="Walk")$
<i>if willing to travel w value</i>	$=IF(!B5="Distance willing to travel")$
<i>access time in minutes</i>	$=(((!F\$6^2)+((B\$32/2)^2))^0.5)/(!D\$6*60))$
<i>actual distance</i>	$=ELSE()$
<i>walk access time in minutes</i>	$=(!H\$6/(!D\$6*60))$
<i>end w value loop</i>	$=END.IF()$
<i>end walk access loop</i>	$=END.IF()$
<i>auto access</i>	$=IF(!B6="Automobile")$
<i>auto access time in minutes</i>	$=!D7+!J7$
<i>end auto access loop</i>	$=END.IF()$
<i>bus access time in minutes</i>	$=IF(!B6="Local Bus/Walk")$
<i>willing to travel w value</i>	$=IF(!B5="Distance willing to travel")$
<i>access time in minutes</i>	$=((((!F\$8^2)+(((B\$32/2)/5280)^2))^0.5)/(!D\$8/60))+(((!P\$8^2)+((B\$32/8)^2))^0.5)/(!N\$8*60))+(!J\$8^0.5)$
<i>actual distance</i>	$=ELSE()$
<i>bus access time in minutes</i>	$=(!H\$8/(!D\$8/60))+(!J\$8^0.5)$

<i>end w value loop</i>	=END.IF()
<i>end bus access loop</i>	=END.IF()
<i>combo access time in minutes</i>	=IF(!B6="Combo of All")
<i>willing to travel w value</i>	=IF(!B5="Distance willing to travel")
<i>combo access time (walk)</i>	=(!L\$6/100)*((((!F\$6^2)+((B\$32/2)^2))^0.5)/(!D\$6*60))
<i>combo access time (auto)</i>	=(!L\$7/100)*(!D\$7+!J\$7)
<i>combo access time (bus)</i>	=(!L\$8/100)*((((!F\$8^2)+(((B\$32/2)/5280)^2))^0.5)/(!D\$8/60))+((((!R\$8^2)+((B\$32/8)^2))^0.5)/(!N\$8*60))+(!J\$8^0.5))
<i>combo access time (total)</i>	=B171+B172+B173
<i>actual distance</i>	=ELSE()
<i>combo access time (walk)</i>	=(!L\$6/100)*((!H\$6^2)/(!D\$20*60))
<i>combo access time (auto)</i>	=(!L\$7/100)*(!D\$7*!J\$7)
<i>combo access time (bus)</i>	=(!L\$8/100)*((!H\$8/(!D\$8/60))+(!J\$8^0.5)+(!R\$8/(!N\$8*60)))
<i>combo access time (total)</i>	=B176+B177+B178
<i>end w value loop</i>	=END.IF()
<i>end combo access loop</i>	=END.IF()
<i>end Typethree access loop</i>	=END.IF()
	=ECHO(FALSE)
<i>no segmentation</i>	=IF(!D2="")
	=ACTIVATE("Summcalc")
	=SELECT(B32)
	=COPY()
	=ACTIVATE("WORKSHEET")
<i>print the spacing on screen</i>	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!A\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE(B4)
<i>Typeone access print loop</i>	=IF(!B\$2=1)
<i>willing to travel w value</i>	=IF(!B\$5="Distance willing to travel")
	=ACTIVATE("Summcalc")
	=SELECT(B\$36)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0))
	=PASTE.SPECIAL(3,1)
<i>actual distance loop</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>print access walk distance</i>	=SELECT(B\$38)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end walk print loop</i>	=END.IF()
<i>end Typeone access print loop</i>	=END.IF()
	=ACTIVATE(B4)

<i>Typeone and three egress loop</i>	=IF(OR(!B\$2=1,!B\$2=3))
<i>walk egress</i>	=IF(OR(!B\$20="Walk",!B\$20="Walk (Adults)",!B\$20="Walk (Elderly)",!B\$20="Walk (Children)",!B\$20="Walk (CBD)"))
<i>if willing to travel w value</i>	=IF(!B5="Distance willing to travel")
<i>print walk w value</i>	=ACTIVATE("Summcalc") =SELECT(B\$44) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0)) =PASTE.SPECIAL(3,1) =ELSE() =ACTIVATE("Summcalc")
<i>print walk distance</i>	=SELECT(B\$46) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end walk loop</i>	=END.IF()
<i>bus egress time in minutes</i>	=ACTIVATE(B4)
<i>willing to travel w value</i>	=IF(!B\$20="Local Bus/Walk") =IF(!B5="Distance willing to travel")
<i>print bus/walk w value</i>	=ACTIVATE("Summcalc") =SELECT(B\$51) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0)) =PASTE.SPECIAL(3,1) =ELSE() =ACTIVATE("Summcalc")
<i>print bus/walk distance</i>	=SELECT(B\$53) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end bus/walk loop</i>	=END.IF()
<i>combo egress time in minutes</i>	=ACTIVATE(B4)
<i>willing to travel w value</i>	=IF(!B\$20="Combo of Both") =IF(!B5="Distance willing to travel")
<i>print combo w value</i>	=ACTIVATE("Summcalc") =SELECT(B\$60)

	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ELSE()
<i>print combo w value</i>	=ACTIVATE("Summcalc")
	=SELECT(B\$64)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end combo loop</i>	=END.IF()
<i>end Typeone or three print loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>Typetwo access and egress loop</i>	=IF(1B\$2=2)
<i>walk access</i>	=IF(1B\$6="Walk")
<i>if willing to travel w value</i>	=IF(1B\$5="Distance willing to travel")
	=ACTIVATE("Summcalc")
<i>access time value</i>	=SELECT(B\$71)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0))
	=PASTE.SPECIAL(3,1)
<i>if w is walk distance</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>walk access time</i>	=SELECT(B\$73)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end walk access loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>auto access</i>	=IF(1B6="Automobile")
	=ACTIVATE("Summcalc")
<i>auto access time</i>	=SELECT(B77)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0))
	=PASTE.SPECIAL(3,1)

<i>end auto access loop</i>	=ACTIVATE("Summcalc") =END.IF()
<i>bus access time in minutes willing to travel w value</i>	=ACTIVATE(B4) =IF(!B\$6="Local Bus/Walk") =IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>access time</i>	=SELECT(B\$81) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0)) =PASTE.SPECIAL(3,1)
<i>if w is actual distance</i>	=ELSE()
<i>bus access time</i>	=ACTIVATE("Summcalc") =SELECT(B\$83) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop end bus access loop</i>	=END.IF() =END.IF()
<i>rail access time in minutes willing to travel w value</i>	=ACTIVATE(B4) =IF(!B\$6="Light Rail/Walk") =IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>rail access time</i>	=SELECT(B\$92) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0)) =PASTE.SPECIAL(3,1)
<i>if w is actual distance</i>	=ELSE()
<i>rail access time</i>	=ACTIVATE("Summcalc") =SELECT(B\$94) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop end rail access loop</i>	=END.IF() =END.IF()
<i>combo access time in minutes willing to travel w value</i>	=ACTIVATE(B4) =IF(!B\$6="Combo of All") =IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")

<i>access time</i>	=SELECT(B\$107) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0)) =PASTE.SPECIAL(3,1)
<i>if w is actual distance</i>	=ELSE() =ACTIVATE("Summcalc")
<i>access wait time</i>	=SELECT(B\$113) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end combo loop</i>	=END.IF() =ACTIVATE(B4)
<i>walk egress loop</i>	=IF(!B\$20="Walk")
<i>if willing to travel w value</i>	=IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>walk time</i>	=SELECT(B\$118) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0)) =PASTE.SPECIAL(3,1)
<i>if w is walk distance</i>	=ELSE() =ACTIVATE("Summcalc")
<i>walk time</i>	=SELECT(B\$120) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end walk egress loop</i>	=END.IF() =ACTIVATE(B4)
<i>bus egress time in minutes</i>	=IF(!B\$20="Local Bus/Walk")
<i>willing to travel w value</i>	=IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>bus,walk time</i>	=SELECT(B\$125) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0)) =PASTE.SPECIAL(3,1)

<i>if w is walk distance</i>	=ELSE()
<i>bus,walk time</i>	=ACTIVATE("Summcalc")
	=SELECT(B\$127)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end bus egress loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>rail egress time in minutes</i>	=IF(!B\$20="Light Rail/Walk")
<i>willing to travel w value</i>	=IF(!B\$5="Distance willing to travel")
	=ACTIVATE("Summcalc")
<i>rail,walk time</i>	=SELECT(B\$132)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
<i>if w is walk distance</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>rail egress time</i>	=SELECT(B\$134)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end rail egress loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>combo egress time in minutes</i>	=IF(!B\$20="Combo of All")
<i>willing to travel w value</i>	=IF(!B\$5="Distance willing to travel")
	=ACTIVATE("Summcalc")
<i>combo egress time</i>	=SELECT(B\$142)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
<i>if w is distance</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>egress time</i>	=SELECT(B\$147)
	=COPY()
	=ACTIVATE("WORKSHEET")

	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end combo egress loop</i>	=END.IF()
<i>end Typetwo loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>Typethree access print loop</i>	=IF(1B\$2=3)
<i>walk access</i>	=IF(1B\$6="Walk")
<i>if willing to travel w value</i>	=IF(1B\$5="Distance willing to travel")
	=ACTIVATE("Summcalc")
	=SELECT(B\$154)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!\$B\$3,count,0))
	=PASTE.SPECIAL(3,1)
<i>if w is walk distance</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>walk access</i>	=SELECT(B\$156)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!\$B\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value loop</i>	=END.IF()
<i>end walk access loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>auto access</i>	=IF(1B\$6="Automobile")
	=ACTIVATE("Summcalc")
<i>access time</i>	=SELECT(B\$160)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end auto access loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>bus access time in minutes</i>	=IF(1B\$6="Local Bus/Walk")
<i>willing to travel w value</i>	=IF(1B\$5="Distance willing to travel")
	=ACTIVATE("Summcalc")
<i>access value</i>	=SELECT(B\$164)
	=COPY()
	=ACTIVATE("WORKSHEET")

	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0))
	=PASTE.SPECIAL(3,1)
<i>if w is actual distance</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>bus access time</i>	=SELECT(B\$166)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end bus access loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>combo access time in minutes</i>	=IF(!B\$6="Combo of All")
<i>willing to travel w value</i>	=IF(!B\$5="Distance willing to travel")
	=ACTIVATE("Summcalc")
<i>access value</i>	=SELECT(B\$174)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0))
	=PASTE.SPECIAL(3,1)
<i>if w is walk distance</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>combo access time</i>	=SELECT(B\$179)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!B\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end combo access loop</i>	=END.IF()
<i>end Typethree access loop</i>	=END.IF()
<i>switch to segmentation print loop</i>	=ELSE()
<i>show end product</i>	=ECHO(FALSE)
	=ACTIVATE("Summcalc")
	=SELECT(B32)
	=COPY()
	=ACTIVATE("WORKSHEETS")
<i>print the spacing on screen</i>	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!B\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=ACTIVATE(B4)
<i>Typeone access loop</i>	=IF(!B\$2=1)

<i>willing to travel w value</i>	=IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc") =SELECT(B36) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>switch to actual distance</i>	=ELSE() =ACTIVATE("Summcalc")
<i>print access walk distance</i>	=SELECT(B\$38) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end Typeone access print loop</i>	=END.IF() =ACTIVATE(B4)
<i>Typeone and three egress loop</i>	=IF(OR(!B\$2=1,!B\$2=3))
<i>walk egress</i>	=IF(OR(!B\$20="Walk",!B\$20="Walk (Adults)",!B\$20="Walk (Elderly)",!B\$20="Walk (Children)",!B\$20="Walk (CBD)"))
<i>if willing to travel w value</i>	=IF(!B5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>print walk w value</i>	=SELECT(B\$44) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc") =ELSE() =ACTIVATE("Summcalc")
<i>print walk distance</i>	=SELECT(B\$46) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end walk loop</i>	=END.IF() =ACTIVATE(B4)
<i>bus egress time in minutes</i>	=IF(!B\$20="Local Bus/Walk")
<i>willing to travel w value</i>	=IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")

<i>print bus/walk w value</i>	=SELECT(B\$51) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1) =ELSE()
<i>print bus/walk distance</i>	=ACTIVATE("Summcalc") =SELECT(B\$53) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end bus/walk loop</i>	=END.IF()
<i>combo egress time in minutes</i>	=ACTIVATE(B4)
<i>willing to travel w value</i>	=IF(!B\$20="Combo of Both") =IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>print combo w value</i>	=SELECT(B\$60) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1) =ELSE()
<i>print combo distance</i>	=ACTIVATE("Summcalc") =SELECT(B\$64) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end combo loop</i>	=END.IF()
<i>end Typeone or three print loop</i>	=END.IF()
<i>Typetwo access and egress loop</i>	=ACTIVATE(B4) =IF(!B\$2=2) =ACTIVATE("Summcalc")
<i>walk access</i>	=IF(!B\$6="Walk")
<i>if willing to travel w value</i>	=IF(!B\$5="Distance willing to travel")
<i>access time value</i>	=SELECT(B\$71) =COPY() =ACTIVATE("WORKSHEETS")

	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer))
	=PASTE.SPECIAL(3,1)
<i>if w is walk distance</i>	=ACTIVATE("Summcalc")
	=ELSE()
	=ACTIVATE("Summcalc")
<i>walk access time</i>	=SELECT(B\$73)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end walk access loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>auto access</i>	=IF(!B6="Automobile")
	=ACTIVATE("Summcalc")
<i>auto access time</i>	=SELECT(B77)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end auto access loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>bus access time in minutes willing to travel w value</i>	=IF(!B\$6="Local Bus/Walk")
	=IF(!B\$5="Distance willing to travel")
	=ACTIVATE("Summcalc")
<i>access time</i>	=SELECT(Summcalc!B\$81)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>if w is actual distance</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>bus access time</i>	=SELECT(B\$83)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end bus access loop</i>	=END.IF()

<i>rail access time in minutes willing to travel w value</i>	=ACTIVATE(B4) =IF(!B\$6="Light Rail/Walk") =IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>rail access time</i>	=SELECT(B\$92) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>if w is actual distance</i>	=ELSE() =ACTIVATE("Summcalc")
<i>rail access time</i>	=SELECT(B\$94) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop end rail access loop</i>	=END.IF() =END.IF() =ACTIVATE(B4)
<i>combo access time in minutes willing to travel w value</i>	=IF(!B6="Combo of All") =IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>access time</i>	=SELECT(B\$107) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>if w is actual distance</i>	=ELSE() =ACTIVATE("Summcalc")
<i>access wait time</i>	=SELECT(B\$113) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop end combo loop</i>	=END.IF() =END.IF() =ACTIVATE(B4)
<i>walk egress loop if willing to travel w value</i>	=IF(!B\$20="Walk") =IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")

<i>walk time</i>	=SELECT(B\$118) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>if w is walk distance</i>	=ELSE() =ACTIVATE("Summcalc")
<i>walk time</i>	=SELECT(B\$120) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end walk egress loop</i>	=END.IF() =ACTIVATE(B4)
<i>bus egress time in minutes</i>	=IF(!B\$20="Local Bus/Walk")
<i>willing to travel w value</i>	=IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>bus,walk time</i>	=SELECT(B\$125) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>if w is walk distance</i>	=ELSE() =SELECT(B\$127)
<i>bus,walk time</i>	=COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end bus egress loop</i>	=END.IF() =ACTIVATE(B4)
<i>rail egress time in minutes</i>	=IF(!B\$20="Light Rail/Walk")
<i>willing to travel w value</i>	=IF(!B\$5="Distance willing to travel")
<i>rail,walk time</i>	=SELECT(B\$132) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer)) =PASTE.SPECIAL(3,1)

<i>if w is walk distance</i>	=ACTIVATE("Summcalc")
	=ELSE()
<i>rail egress time</i>	=ACTIVATE("Summcalc")
	=SELECT(B\$134)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS!F\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end rail egress loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>combo egress time in minutes</i>	=IF(!B\$20="Combo of All")
<i>willing to travel w value</i>	=IF(!B\$5="Distance willing to travel")
	=ACTIVATE("Summcalc")
<i>combo egress time</i>	=SELECT(B\$142)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS!F\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>if w is distance</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>egress time</i>	=SELECT(B\$147)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS!F\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end combo egress loop</i>	=END.IF()
<i>end Typetwo loop</i>	=END.IF()
	=ACTIVATE(B4)
<i>Typethree access print loop</i>	=IF(!B\$2=3)
<i>walk access</i>	=IF(!B\$6="Walk")
<i>if willing to travel w value</i>	=IF(!B\$5="Distance willing to travel")
	=ACTIVATE("Summcalc")
	=SELECT(B\$154)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS!\$C\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>if w is walk distance</i>	=ELSE()

<i>walk access</i>	=ACTIVATE("Summcalc") =SELECT(B\$156) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$C\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value loop</i>	=END.IF()
<i>end walk access loop</i>	=END.IF()
<i>auto access</i>	=ACTIVATE(B4) =IF(!B\$6="Automobile") =ACTIVATE("Summcalc")
<i>access time</i>	=SELECT(B\$160) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$C\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end auto access loop</i>	=END.IF()
<i>bus access time in minutes willing to travel w value</i>	=ACTIVATE(B4) =IF(!B\$6="Local Bus/Walk") =IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>access value</i>	=SELECT(B\$164) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$C\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>if w is actual distance</i>	=ELSE() =ACTIVATE("Summcalc")
<i>bus access time</i>	=SELECT(B\$166) =COPY() =ACTIVATE("WORKSHEETS") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$C\$3,count,answer)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end bus access loop</i>	=END.IF()
<i>combo access time in minutes willing to travel w value</i>	=ACTIVATE(B4) =IF(!B\$6="Combo of All") =IF(!B\$5="Distance willing to travel") =ACTIVATE("Summcalc")
<i>access value</i>	=SELECT(B\$174)

	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$C\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>if w is walk distance</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>combo access time</i>	=SELECT(B\$179)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$C\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>end w value or distance loop</i>	=END.IF()
<i>end combo access loop</i>	=END.IF()
<i>end Typethree access loop</i>	=END.IF()
<i>end segment print loop</i>	=END.IF()
	=ACTIVATE(B4)
	=MESSAGE(TRUE,"Computing the waiting time cost.")
<i>half HW waiting time</i>	=IF(!F2=".5*HEADWAY")
	=!F3/2
	=END.IF()
<i>sq root HW waiting time</i>	=IF(!F2="(HEADWAY)^.5")
	=!F3^0.5
	=END.IF()
<i>30 min split waiting time less than 30 min HW</i>	=IF(!F2=".5 if HW>30,^.5 if HW<30")
	=IF(!F3<30)
	=!F3/2
<i>more than 30 min HW</i>	=ELSE()
	=!F3^0.5
	=END.IF()
	=END.IF()
	=ECHO(FALSE)
	=ACTIVATE(B4)
<i>print waiting time (no seg)</i>	=IF(!D2="")
<i>half HW waiting time</i>	=IF(!F\$2=".5*HEADWAY")
	=ACTIVATE("Summcalc")
<i>print wait time</i>	=SELECT(B\$837)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!C\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
	=ACTIVATE(B4)

<i>sq root HW waiting time</i>	=IF(!F\$2="(HEADWAY)^.5")
<i>print wait time</i>	=ACTIVATE("Summcalc")
	=SELECT(B\$840)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!C\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
	=ACTIVATE(B4)
<i>30 min split waiting time less than 30 min HW</i>	=IF(!F\$2=".5 if HW>30,^.5 if HW<30")
	=IF(!F\$3<30)
	=ACTIVATE("Summcalc")
<i>print wait time</i>	=SELECT(B\$844)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!C\$3,count,0))
	=PASTE.SPECIAL(3,1)
<i>greater than 30 min</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>print wait time</i>	=SELECT(B\$846)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!C\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
	=END.IF()
	=ACTIVATE(B4)
<i>user specified waiting time</i>	=IF(!F\$2="USER SPECIFIED")
	=ACTIVATE(B4)
<i>print wait time</i>	=SELECT(!B10)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!C\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
<i>switch to segmentation</i>	=ELSE()
	=ACTIVATE(B4)
<i>half HW waiting time</i>	=IF(!F\$2=".5*HEADWAY")
	=ACTIVATE("Summcalc")
<i>print wait time</i>	=SELECT(B\$837)
	=COPY()

	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!D\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
	=ACTIVATE(B4)
<i>sq root HW waiting time</i>	=IF(!F\$2="(HEADWAY)^.5")
	=ACTIVATE("Summcalc")
<i>print wait time</i>	=SELECT(B\$840)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!D\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
	=ACTIVATE(B4)
<i>30 min split waiting time less than 30 min HW</i>	=IF(!F\$2=".5 if HW>30,^.5 if HW<30")
	=IF(!F\$3<30)
	=ACTIVATE("Summcalc")
<i>print wait time</i>	=SELECT(B\$844)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!D\$3,count,answer))
	=PASTE.SPECIAL(3,1)
<i>greater than 30 min</i>	=ELSE()
	=ACTIVATE("Summcalc")
<i>print wait time</i>	=SELECT(B\$846)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!D\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
	=END.IF()
	=ACTIVATE(B4)
<i>user specified waiting time</i>	=IF(!F\$2="USER SPECIFIED")
	=ACTIVATE(B4)
<i>print wait time</i>	=SELECT(!B10)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!D\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")

<i>end waiting time print loop</i>	=END.IF()
<i>increment spacing counter</i>	=END.IF() =SET.NAME("count", count+1)
<i>return to test condition</i>	=ACTIVATE(B4) =NEXT() =ACTIVATE(B4)
<i>Linehaul calculations (no seg)</i>	=MESSAGE(TRUE,"Computing the in-vehicle cost.")
<i>figure Poisson dwell time</i>	=IF(ID2="")
<i>initialize counter to zero</i>	=IF(ID10="") =SET.NAME("count",0)
<i>test conditions to continue loop</i>	=ACTIVATE(B4) =WHILE(OR(!E\$4+(count*!C\$4)<!G\$4,!E\$4+(count*! C\$4)=!G\$4))
<i>display the spacing</i>	=ACTIVATE(B4) =!E\$4+(!C\$4*count)
<i>accel distance</i>	=(!J\$10^2)/(2*!F\$10)
<i>accel time (min)</i>	=((!J\$10/!F\$10)/60)
<i>decel distance</i>	=(!J\$10^2)/(2*!H\$10)
<i>decel time (min)</i>	=((!J\$10/!H\$10)/60)
<i>accel+decel dist (ft)</i>	=B963+B965
<i>accel+decel time (min)</i>	=B964+B966
<i>cruise distance</i>	=B962-B967
<i>avg trip length</i>	=!H\$3*5280
<i>#cycles for avg trip length</i>	=B970/B962
<i>route length</i>	=!J\$3*5280
<i>#cycles for route length</i>	=B972/B962
<i>fractional cycle avg trip</i>	=B971-INT(B971)
<i>distance covered in frac cycle</i>	=B974*B962
<i>fractional cycle route length</i>	=B973-INT(B973)
<i>distance covered in frac cycle</i>	=B976*B962
<i>figure dwell time (poss stops)</i>	=INT(((!J\$3*5280)/B962))
<i>total board in zone per min</i>	=(!C\$11/60)/B978
<i>prob of at least one board pax</i>	=(1-(2.718^(-B979*!F\$3)))
<i># actual stops if Pois arr</i>	=B978*B980
<i>find pax/stop (Q)</i>	
<i>mean arrival rate</i>	=((!C\$11/60)/B981)*!F\$3
<i>begin counter</i>	=SET.NAME("dog",0)
<i>begin loop</i>	=FOR("dog",0,50,1) =dog
<i>probability of x boarding pax</i>	=POISSON(B986,B983,TRUE)
<i>is cum prob less than desired</i>	=IF(B987>(!G\$12/100),BREAK()) =NEXT()
<i>probability</i>	=B987
<i># pax for that prob (pax/stop)</i>	=dog
<i>board time in zone (sec)</i>	=(B981*!C\$12)+(B991*B981*!G\$11)
<i>#actual stops(alight)</i>	
<i>total alight in zone per min</i>	=(!E\$11/60)/B978

<i>prob of at least one alight pax</i>	=(1-(2.718^(-B994*!F\$3)))
<i># actual stops if Pois arr</i>	=B978*B995
find pax/stop (Q)	
<i>mean arrival rate</i>	=(!(E\$11/60)/B996)*!F\$3
<i>begin counter</i>	=SET.NAME("cat",0)
<i>begin loop</i>	=FOR("cat",0,50,1)
	=cat
<i>prob of x alighting pax</i>	=POISSON(B1001,B998,TRUE)
<i>is cum prob less than desired</i>	=IF(B1002>(IG\$12/100),BREAK())
<i>recalculate poisson prob</i>	=NEXT()
<i>probability</i>	=B1002
<i>#pax for that prob (pax/stop)</i>	=cat
alight time in zone (sec)	=(B996*!E\$12)+(B1006*B996*!G\$11)
<i>total # stops for both</i>	=(B978*((1-(2.718~(B983+B998))))))
<i>max time board, alight</i>	=MAX(B992,B1007)
<i>dwel time per stop (sec)</i>	=B1009/B1008
test if cruise dist is neg.	=IF(B969<0)
<i>ratio of decel to accel</i>	=!H\$10!/F\$10
<i>time to accel/decel (min)</i>	=(2*B962)/((B1012+1)*!F\$10)^0.5/60
<i>one full cycle time(min)</i>	=B1013+(B1010/60)
<i>frac cycle time avg trip (min)</i>	=(2*ABS(B975))/((B1012+1)*!F\$10)^0.5/60
<i>frac cycle time route seg (min)</i>	=(2*ABS(B977))/((B1012+1)*!F\$10)^0.5/60
<i>on board time avg seg (min)</i>	=(B1014*(INT(B971)))+B1015
<i>on board time route seg (min)</i>	=(B1014*(INT(B973)))+B1016
	=END.IF()
test cr dist= acc/dec dist	=IF(B969=0)
<i>ratio of decel to accel</i>	=!H\$10!/F\$10
<i>one full cycle time(min)</i>	=B968+(B1010/60)
<i>frac cycle time avg trip (min)</i>	=(2*ABS(B975))/((B1021+1)*!F\$10)^0.5/60
<i>frac cycle time route seg (min)</i>	=(2*ABS(B977))/((B1021+1)*!F\$10)^0.5/60
<i>on board time avg trip (min)</i>	=(B1022*(INT(B971)))+B1023
<i>on board time route seg (min)</i>	=(B1022*(INT(B973)))+B1024
	=END.IF()
test if cruise dist is pos	=IF(B969>0)
<i>cruise time (min)</i>	=(B969/IJ\$10)/60
<i>one full cycle time(min)</i>	=B968+B1029+(B1010/60)
<i>test frac can reach cruise (avg)</i>	=IF(B975>B967)
<i>time to cruise frac (avg)(min)</i>	=(B975/IJ\$10)/60
<i>on board time avg trip (min)</i>	=(B1030*(INT(B971)))+B1032
	=ELSE()
<i>ratio of decel to accel</i>	=!F\$10!/H\$10
<i>frac cycle time avg trip (min)</i>	=(2*ABS(B975))/((B1035+1)*!F\$10)^0.5/60
<i>on board time avg trip (min)</i>	=(B1030*(INT(B971)))+B1036
	=END.IF()
<i>test frac can reach cruise (rte)</i>	=IF(B977>B967)
<i>time to cruise frac (rte)(min)</i>	=(B977/IJ\$10)/60
<i>on board time route len (min)</i>	=(B1030*(INT(B973)))+B1040

<i>ratio of decel to accel</i>	=ELSE()
<i>frac cycle time route len (min)</i>	=!F\$10!/H\$10
<i>on board time route len (min)</i>	=((2*ABS(B977))/((B1043+1)*!F\$10))^0.5/60
	=(B1030*(INT(B973)))+B1044
	=END.IF()
<i>end pos cruise loop</i>	=END.IF()
	=ECHO(FALSE)
	=MESSAGE(TRUE,"Recording the in-vehicle costs.")
<i>Linehaul print loop</i>	=IF(B969<0)
<i>if cruise dist is neg</i>	=SELECT(B1017)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!D\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=SELECT(B1018)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!E\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
<i>if cr dist= acc/dec dist</i>	=IF(B969=0)
<i>if cruise dist is neg</i>	=SELECT(B1025)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!D\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=SELECT(B1026)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!E\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
<i>if cruise dist is pos and cruise cruise reached in frac avg trip</i>	=IF(B969>0)
	=IF(B975>B967)
	=SELECT(B1033)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!D\$3,count,0))
	=PASTE.SPECIAL(3,1)

<i>if cruise dist is pos and cruise can't be reached in frac cycle avg trip seg</i>	=ACTIVATE("Summcalc") =ELSE() =ACTIVATE("Summcalc") =SELECT(B1037) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!D\$3,count,0)) =PASTE.SPECIAL(3,1) =END.IF()
<i>cruise reached in frac cycle route seg and cr dist pos</i>	=ACTIVATE("Summcalc") =IF(B977>B967) =SELECT(B1041) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!E\$3,count,0)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>if cruise dist is pos and cruise can't be reached in frac cycle route seg</i>	=ELSE() =ACTIVATE("Summcalc") =SELECT(B1045) =COPY() =ACTIVATE("WORKSHEET") =SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!E\$3,count,0)) =PASTE.SPECIAL(3,1) =ACTIVATE("Summcalc")
<i>end cruise is pos loop increment counter</i>	=END.IF() =END.IF() =SET.NAME("count",count+1) =ACTIVATE(B4)
<i>return to test condition switch to given dwell time initialize counter to zero</i>	=NEXT() =ELSE() =SET.NAME("count",0) =MESSAGE(TRUE,"Computing the in-vehicle costs.") =ACTIVATE(B4)
<i>test conditions to continue loop</i>	=WHILE(OR(!E\$4+count*!C\$4<!G\$4,!E\$4+count*!C \$4=!G\$4))
<i>display the spacing</i>	=!E\$4+(!C\$4*count)
<i>accel distance</i>	=(!J\$10^2)/(2*!F\$10)
<i>accel time (min)</i>	=((!J\$10/!F\$10)/60)
<i>decel distance</i>	=(!J\$10^2)/(2*!H\$10)
<i>decel time (min)</i>	=((!J\$10/!H\$10)/60)
<i>accel+decel dist (ft)</i>	=B1121+B1123
<i>accel+decel time (min)</i>	=B1122+B1124
<i>cruise distance</i>	=B1120-B1125
<i>avg trip length</i>	=!H\$3*5280

```

#cycles for avg trip length      =B1128/B1120
route length                    =!J$3*5280
#cycles for route length        =B1130/B1120
fractional cycle avg trip       =B1129-INT(B1129)
distance covered in frac cycle  =B1132*B1120
fractional cycle route length   =B1131-INT(B1131)
distance covered in frac cycle  =B1134*B1120
test if cruise dist is neg.    =IF(B1127<0)
ratio of decel to accel        =!H$10!/F$10
time to accel/decel (min)      =(((2*B1120)/((B1137+1)*!F$10))^0.5)/60
one full cycle time(min)       =B1138+(!D10/60)
frac cycle time avg trip (min)  =(((2*ABS(B1133))/((B1137+1)*!F$10))^0.5)/60
frac cycle time route len (min) =(((2*ABS(B1135))/((B1137+1)*!F$10))^0.5)/60
on board time avg trip (min)    =(B1139*(INT(B1129)))+B1140
on board time route len (min)  =(B1139*(INT(B1131)))+B1141
                                =END.IF()
test cr dist= acc/dec dist   =IF(B1127=0)
ratio of decel to accel        =!H$10!/F$10
one full cycle time(min)       =B1126+(!D10/60)
frac cycle time avg trip (min)  =(((2*ABS(B1133))/((B1146+1)*!F$10))^0.5)/60
frac cycle time route len (min) =(((2*ABS(B1135))/((B1146+1)*!F$10))^0.5)/60
on board time avg trip (min)    =(B1147*(INT(B1129)))+B1148
on board time route seg (min)  =(B1147*(INT(B1131)))+B1149
                                =END.IF()
test if cruise dist is pos   =IF(B1127>0)
cruise time (min)              =(B1153!/J$10)/60
one full cycle time(min)       =B1126+B1154+(!D$10/60)
test frac can reach cruise (avg) =IF(B1133>B1125)
time to cruise frac (avg)( min) =(B1133!/J$10)/60
on board time avg trip (min)    =(B1155*(INT(B1129)))+B1157
                                =ELSE()
ratio of decel to accel        =!F$10!/H$10
frac cycle time avg trip (min)  =(((2*ABS(B1133))/((B1160+1)*!F$10))^0.5)/60
on board time avg trip (min)    =(B1155*(INT(B1129)))+B1161
                                =END.IF()
test frac can reach cruise (rte) =IF(B1135>B1125)
time to cruise frac (rte)( min) =(B1135!/J$10)/60
on board time route len (min)  =(B1155*(INT(B1131)))+B1165
                                =ELSE()
ratio of decel to accel        =!F$10!/H$10
frac cycle time route len (min) =(((2*ABS(B1135))/((B1168+1)*!F$10))^0.5)/60
on board time route len (min)  =(B1155*(INT(B1131)))+B1169
                                =END.IF()
end pos cruise loop            =END.IF()
                                =ECHO(FALSE)
                                =MESSAGE(TRUE,"Recording the in-vehicle costs.")
Linehaul print loop            =IF(B1127<0)

```

if cruise dist is neg

```
=SELECT(B1142)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET'!$D$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Summcalc")
=SELECT(B1143)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET'!$E$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Summcalc")
=END.IF()
```

*if cr dist= acc/dec dist
if cruise dist is neg*

```
=ACTIVATE("Summcalc")
=IF(B1127=0)
=SELECT(B1150)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET'!$D$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Summcalc")
=SELECT(B1151)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET'!$E$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Summcalc")
=END.IF()
```

*if cruise dist is pos and cruise
cruise reached in frac avg trip*

```
=ACTIVATE("Summcalc")
=IF(B1127>0)
=IF(B1133>B1125)
=SELECT(B1158)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET'!D$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Summcalc")
=ELSE()
```

*if cruise dist is pos and cruise
can't be reached in frac cycle
avg trip seg*

```
=ACTIVATE("Summcalc")
=SELECT(B1162)
=COPY()
=ACTIVATE("WORKSHEET")
```

	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!D\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=END.IF()
<i>cruise reached in frac cycle route seg and cr dist pos</i>	=ACTIVATE("Summcalc")
	=IF(B1135>B1125)
	=SELECT(B1166)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!E\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>if cruise dist is pos and cruise can't be reached in frac cycle route seg</i>	=ELSE()
	=ACTIVATE("Summcalc")
	=SELECT(B1170)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!E\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
<i>end cruise is pos loop increment counter</i>	=END.IF()
	=SET.NAME("count",count+1)
	=ACTIVATE(B4)
<i>return to test condition end given dwell time loop switch to segmentation loop</i>	=NEXT()
	=END.IF()
	=ELSE()
<i>figure Poisson dwell time initialize counter to zero</i>	=MESSAGE(TRUE,"Computing the in-vehicle costs.")
	=IF(!D10="")
	=SET.NAME("count",0)
	=ACTIVATE(B4)
<i>spacing increment variable test conditions to continue loop</i>	=(OFFSET(!K\$12,seg,0))
	=WHILE(OR(!E\$4+count*B1248<!G\$4,!E\$4+count*B 1248=!G\$4))
<i>display the spacing velocity variable</i>	=!E\$4+(B1248*count)
<i>accel distance</i>	=(OFFSET(!I\$12,seg,0))
<i>accel time (min)</i>	=(B1251^2)/(2*!F\$10)
<i>decel distance</i>	=(B1251!/F\$10)/60
<i>decel time (min)</i>	=(B1251^2)/(2*!H\$10)
<i>accel+decel dist (ft)</i>	=(B1251!/H\$10)/60
<i>accel+decel time (min)</i>	=B1252+B1254
<i>cruise distance</i>	=B1253+B1255
<i>segment length variable</i>	=B1250-B1256
<i>segment length in feet</i>	=(OFFSET(!C\$12,seg,0))
<i>#cycles for segment len</i>	=B1259*5280
	=B1260/B1250

<i>fractional cycle segment</i>	=B1261-INT(B1261)
<i>dist covered in frac cycle</i>	=B1262*B1250
<i>boarding demand variable</i>	=(OFFSET(!E\$12,seg,0))
<i>deboarding demand variable</i>	=(OFFSET(!G\$12,seg,0))
<i>figure dwell time (poss stops)</i>	=INT(((I\$3*5280)/B1250))
<i>total board in zone per min</i>	=(B1264/60)/B1266
<i>prob of at least one board pax</i>	=(1-(2.718^(-B1267*!F\$3)))
<i># actual stops if Pois arr</i>	=B1266*B1268
find pax/stop (Q)	
<i>mean arrival rate</i>	=((B1264/60)/B1269)*!F\$3
<i>begin counter</i>	=SET.NAME("dog",0)
<i>begin loop</i>	=FOR("dog",0,50,1)
	=dog
<i>probability of x boarding pax</i>	=POISSON(B1274,B1271,TRUE)
<i>is cum prob less than desired</i>	=IF(B1275>(!G\$12/100),BREAK())
	=NEXT()
<i>probability</i>	=B1275
<i># pax for that prob (pax/stop)</i>	=dog
board time in zone (sec)	=(B1269*!C\$12)+(B1279*B1269*!G\$11)
#actual stops(alight)	
<i>total alight in zone per min</i>	=(B1265/60)/B1266
<i>prob of at least one alight pax</i>	=(1-(2.718^(-B1282*!F\$3)))
<i># actual stops if Pois arr</i>	=B1266*B1283
find pax/stop (Q)	
<i>mean arrival rate</i>	=(B1265/60)/B1284
<i>begin counter</i>	=SET.NAME("cat",0)
<i>begin loop</i>	=FOR("cat",0,50,1)
	=cat
<i>prob of x alighting pax</i>	=POISSON(B1289,B1286,TRUE)
<i>is cum prob less than desired</i>	=IF(B1290>(!G\$12/100),BREAK())
<i>recalculate poisson prob</i>	=NEXT()
<i>probability</i>	=B1290
<i>#pax for that prob (pax/stop)</i>	=cat
alight time in zone (sec)	=(B1284*!E\$12)+(B1294*B1284*!G\$11)
<i>total # stops for both</i>	=(B1266*((1-(2.718^(-B1271+B1286))))))
<i>max time board, alight</i>	=MAX(B1280,B1295)
<i>dwell time per stop (sec)</i>	=B1297/B1296
test if cruise dist is neg.	=IF(B1258<0)
<i>ratio of decel to accel</i>	=!H\$10/!F\$10
<i>time to accel/decel (min)</i>	=(2*B1250)/((B1300+1)*!F\$10)^0.5/60
<i>one full cycle time(min)</i>	=B1301+(B1298/60)
<i>frac cycle time seg (min)</i>	=(2*ABS(B1263))/((B1300+1)*!F\$10)^0.5/60
<i>on board time seg (min)</i>	=(B1302*(INT(B1262)))+B1303
	=END.IF()
test cr dist= acc/dec dist	=IF(B1258=0)
<i>ratio of decel to accel</i>	=!H\$10/!F\$10
<i>one full cycle time(min)</i>	=B1257+(B1298/60)

<i>frac cycle time segment (min)</i>	=(((2*ABS(B1263))/((B1307+1)*!F\$10))^0.5)/60
<i>on board time segment (min)</i>	=(B1308*(INT(B1262)))+B1309
	=END.IF()
test if cruise dist is pos	=IF(B1258>0)
<i>cruise time (min)</i>	=(B1258/B1251)/60
<i>one full cycle time(min)</i>	=B1257+B1313+(B1298/60)
<i>test frac can reach cruise (seg)</i>	=IF(B1263>B1256)
<i>time to cruise frac (seg)(min)</i>	=(B1263/B1251)/60
<i>on board time seg length (min)</i>	=(B1314*(INT(B1262)))+B1316
	=ELSE()
<i>ratio of decel to accel</i>	=IH\$10/!F\$10
<i>frac cycle time seg length (min)</i>	(((2*ABS(B1263))/((B1319+1)*!F\$10))^0.5)/60
<i>on board time seg length (min)</i>	=(B1314*(INT(B1262)))+B1320
<i>end fractional loop</i>	=END.IF()
<i>end pos cru dist loop</i>	=END.IF()
	=ECHO(FALSE)
	=MESSAGE(TRUE,"Recording the in-vehicle costs.")
	=ACTIVATE("Summcalc")
<i>Linehaul print loop</i>	=IF(B1258<0)
<i>if cruise dist is neg</i>	=SELECT(B1304)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$E\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
	=ACTIVATE("Summcalc")
<i>if cr dist= acc/dec dist</i>	=IF(B1258=0)
	=SELECT(B1310)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$E\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
	=ACTIVATE("Summcalc")
<i>if cruise dist is pos</i>	=IF(B1258>0)
<i>cruise reached in frac avg trip</i>	=IF(B1263>B1256)
	=SELECT(B1317)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$E\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>if cruise dist is pos and cruise</i>	=ELSE()

<i>can't be reached in frac cycle</i>	=ACTIVATE("Summcalc")
<i>avg trip seg</i>	=SELECT(B1321)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!E\$3,count,answer))
	=PASTE.SPECIAL(3,1)
<i>end fractional cruise loop</i>	=END.IF()
<i>end cruise is pos loop</i>	=END.IF()
<i>increment counter</i>	=SET.NAME("count",count+1)
	=ACTIVATE(B4)
<i>return to test condition</i>	=NEXT()
<i>switch to given dwell time</i>	=ELSE()
	=MESSAGE(TRUE,"Computing the in-vehicle costs.")
<i>initialize counter to zero</i>	=SET.NAME("count",0)
	=ACTIVATE(B4)
<i>spacing increment variable</i>	=(OFFSET(!K\$12,seg,0))
<i>test conditions to continue loop</i>	=WHILE(OR(!E\$4+count*B1369<!G\$4,!E\$4+count*B 1369=!G\$4))
<i>display the spacing</i>	=!E\$4+(B1369*count)
<i>velocity variable</i>	=(OFFSET(!I\$12,seg,0))
<i>accel distance</i>	=(B1372^2)/(2*!F\$10)
<i>accel time (min)</i>	=(B1372/!F\$10)/60
<i>decel distance</i>	=(B1372^2)/(2*!H\$10)
<i>decel time (min)</i>	=(B1372/!H\$10)/60
<i>accel+decel dist (ft)</i>	=B1373+B1375
<i>accel+decel time (min)</i>	=B1374+B1376
<i>cruise distance</i>	=B1371-B1377
<i>segment length variable</i>	=(OFFSET(!C\$12,seg,0))
<i>segment length in feet</i>	=B1380*5280
<i>#cycles for segment len</i>	=B1381/B1371
<i>fractional cycle segment</i>	=B1382-INT(B1382)
<i>dist covered in frac cycle</i>	=B1383*B1371
test if cruise dist is neg.	=IF(B1379<0)
<i>ratio of decel to accel</i>	=!H\$10/!F\$10
<i>time to accel/decel (min)</i>	=(((2*B1371)/((B1386+1)*!F\$10))^0.5)/60
<i>one full cycle time(min)</i>	=B1387+(!D\$10/60)
<i>frac cycle time seg (min)</i>	=(((2*ABS(B1382))/((B1386+1)*!F\$10))^0.5)/60
<i>on board time seg (min)</i>	=(B1388*(INT(B1384)))+B1389
	=END.IF()
test cr dist= acc/dec dist	=IF(B1379=0)
<i>ratio of decel to accel</i>	=!H\$10/!F\$10
<i>one full cycle time(min)</i>	=B1378+(!D\$10/60)
<i>frac cycle time segment (min)</i>	=(((2*ABS(B1382))/((B1393+1)*!F\$10))^0.5)/60
<i>on board time segment (min)</i>	=(B1394*(INT(B1384)))+B1395
	=END.IF()
test if cruise dist is pos	=IF(B1379>0)

<i>cruise time (min)</i>	=(B1379/B1372)/60
<i>one full cycle time(min)</i>	=B1378+B1399+(!D\$10/60)
<i>test frac can reach cruise (seg)</i>	=IF(B1382>B1377)
<i>time to cruise frac (seg)(min)</i>	=(B1382/B1372)/60
<i>on board time seg length (min)</i>	=(B1400*(INT(B1384)))+B1402
	=ELSE()
<i>ratio of decel to accel</i>	=!H\$10!/F\$10
<i>frac cycle time seg length (min)</i>	=(2*ABS(B1382))/((B1405+1)*!F\$10)^0.5/60
<i>on board time seg length (min)</i>	=(B1400*(INT(B1384)))+B1406
<i>end fractional loop</i>	=END.IF()
<i>end pos cru dist loop</i>	=END.IF()
	=ECHO(FALSE)
	=MESSAGE(TRUE,"Recording the in-vehicle costs.")
	=ACTIVATE("Summcalc")
<i>Linehaul print loop</i>	=IF(B1379<0)
<i>if cruise dist is neg</i>	=SELECT(B1390)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$E\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
	=ACTIVATE("Summcalc")
<i>if cr dist= acc/dec dist</i>	=IF(B1379=0)
	=SELECT(B1396)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!\$E\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
	=END.IF()
	=ACTIVATE("Summcalc")
<i>if cruise dist is pos</i>	=IF(B1379>0)
<i>cruise reached in frac avg trip</i>	=IF(B1379>B1372)
	=SELECT(B1403)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!E\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Summcalc")
<i>if cruise dist is pos and cruise can't be reached in frac cycle avg trip seg</i>	=ELSE()
	=ACTIVATE("Summcalc")
	=SELECT(B1407)
	=COPY()
	=ACTIVATE("WORKSHEETS")

*end fractional cruise loop
end cruise is pos loop
increment counter*

*return to test condition
end dwell time loop
end segmentation loop
increment column counter
increment seg counter*

return to multiple segments

*don't incr counter if one seg
clear zeroes from results*

clear zeroes from results

return control to Backbone

```
=SELECT(OFFSET('JAD HD:REAL  
DEAL:WORKSHEETS'!E$3,count,answer))  
=PASTE.SPECIAL(3,1)  
=END.IF()  
=END.IF()  
=SET.NAME("count",count+1)  
=ACTIVATE(B4)  
=MESSAGE(TRUE,"Incrementing the loop counter.")  
=NEXT()  
=END.IF()  
=END.IF()  
=SET.NAME("answer",answer+8)  
=SET.NAME("seg",seg+1)  
=ACTIVATE(B4)  
=NEXT()  
=ACTIVATE(B4)  
=MESSAGE(FALSE)  
=IF(ID2="")  
=ACTIVATE("WORKSHEET")  
=SELECT('JAD HD:REAL DEAL:WORKSHEET'!$G$3:$I$25)  
=FOR.CELL("CURRENT",,TRUE)  
= CURRENT  
= IF(CURRENT=0)  
= FORMULA(" ",CURRENT)  
= END.IF()  
=NEXT()  
=ELSE()  
=ACTIVATE("WORKSHEETS")  
=SELECT('JAD HD:REAL DEAL:WORKSHEETS'!G3:AV25)  
=FOR.CELL("CURRENT",,TRUE)  
= CURRENT  
= IF(CURRENT=0)  
= FORMULA(" ",CURRENT)  
= END.IF()  
=NEXT()  
=END.IF()  
=MESSAGE(FALSE)  
=RETURN()
```

Typeone Subroutine

<i>name</i>	Typeone =ECHO(FALSE) =MESSAGE(TRUE,"Opening the necessary files.") =OPEN("JAD HD:REAL DEAL:opener",0,TRUE)
<i>open files;changes not saved</i>	=RUN('JAD HD:REAL DEAL:opener'!B\$2) =ACTIVATE("Typeone") =MESSAGE(FALSE)
<i>name to save results</i>	=INPUT("Enter a name under which to save the results from this run.",2,"RESULTS") =INPUT("Enter the spacing increment (ft):",1,"SPACING INCREMENT",400) =IF(OR(B9<0,B9=0))
<i>input operating policy type</i>	=ALERT("The increment must be a number greater than 0.",3) =GOTO(B9) =END.IF() =SELECT(B9) =COPY() =ACTIVATE("Summary") =SELECT('JAD HD:REAL DEAL:Summary'!C4)
<i>print the spacing on screen</i>	=PASTE.SPECIAL(3,1) =ACTIVATE("Typeone")
<i>spacing inputs for the loop</i>	=INPUT("Enter the mininum spacing (ft):",1,"MINIMUM SPACING",400) =IF(OR(B20<0,B20=0))
<i>input operating policy type</i>	=ALERT("The spacing must be a number greater than 0.",3) =GOTO(B20) =END.IF() =SELECT(B20) =COPY() =ACTIVATE("Summary") =SELECT('JAD HD:REAL DEAL:Summary'!E4)
<i>print the spacing on screen</i>	=PASTE.SPECIAL(3,1) =ACTIVATE("Typeone") =INPUT("Enter the maximum spacing (ft):",1,"MAXIMUM SPACING",6000) =IF(OR(B31<0,B31=0))
<i>input operating policy type</i>	=ALERT("The spacing must be a number greater than 0.",3) =GOTO(B31) =END.IF() =SELECT(B31) =COPY() =ACTIVATE("Summary") =SELECT('JAD HD:REAL DEAL:Summary'!G4)
<i>print the spacing on screen</i>	=PASTE.SPECIAL(3,1) =ACTIVATE("Typeone")

*turn off screen updating
linehaul inputs
negative number loop*

```
=ECHO(FALSE)  
=DIALOG.BOX('JAD HD:REAL DEAL:LHM1'!B3:H22)  
=IF(OR('JAD HD:REAL DEAL:LHM1'!H7<0,'JAD HD:REAL  
DEAL:LHM1'!H9<0,'JAD HD:REAL DEAL:LHM1'!H11<0,'JAD  
HD:REAL DEAL:LHM1'!H13<0,'JAD HD:REAL  
DEAL:LHM1'!H15<0,'JAD HD:REAL DEAL:LHM1'!H17<0,'JAD  
HD:REAL DEAL:LHM1'!H20<0))  
=ALERT("The inputs must be positive numbers.",2)  
=GOTO(Typeone!B43)  
=END.IF()
```

print line haul mode

```
=MESSAGE(TRUE,"Recording the Linehaul inputs to the  
summary statistics file.")  
=ACTIVATE("LHM1")  
=SELECT('JAD HD:REAL DEAL:LHM1'!H5)  
=COPY()  
=ACTIVATE("Summary")  
=SELECT('JAD HD:REAL DEAL:Summary'!B3)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("LHM1")  
=SELECT('JAD HD:REAL DEAL:LHM1'!H7)  
=COPY()
```

print bandwidth

```
=ACTIVATE("Summary")  
=SELECT('JAD HD:REAL DEAL:Summary'!D3)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("LHM1")  
=SELECT('JAD HD:REAL DEAL:LHM1'!H9)  
=COPY()  
=ACTIVATE("Summary")
```

print route headway

```
=SELECT('JAD HD:REAL DEAL:Summary'!F3)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("LHM1")  
=SELECT('JAD HD:REAL DEAL:LHM1'!H11)  
=COPY()  
=ACTIVATE("Summary")
```

print accel rate

```
=SELECT('JAD HD:REAL DEAL:Summary'!F10)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("LHM1")  
=SELECT('JAD HD:REAL DEAL:LHM1'!H13)  
=COPY()  
=ACTIVATE("Summary")
```

print decel rate

```
=SELECT('JAD HD:REAL DEAL:Summary'!H10)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("LHM1")  
=SELECT('JAD HD:REAL DEAL:LHM1'!H17)  
=COPY()  
=ACTIVATE("Summary")
```

print cruise velocity

```
=SELECT('JAD HD:REAL DEAL:Summary'!J10)  
=PASTE.SPECIAL(3,1)
```

```

=ACTIVATE("LHM1")
=SELECT('JAD HD:REAL DEAL:LHM1'!H17)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary'!H3)
= PASTE.SPECIAL(3,1)
=ACTIVATE("LHM1")
=SELECT('JAD HD:REAL DEAL:LHM1'!H20)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary'!J3)
= PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=MESSAGE(FALSE)
=ECHO(FALSE)
w value preference
=DIALOG.BOX('JAD HD:REAL DEAL:WVAL1'!B3:H8)
=ACTIVATE("WVAL1")
=SELECT('JAD HD:REAL DEAL:WVAL1'!H5)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary'!B5)
print to summary sheet
= PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=IF('JAD HD:REAL DEAL:WVAL1'!H6=1)
=MESSAGE(TRUE,"Opening the necessary files.")
=ACTIVATE("opener")
open files;changes not saved
=RUN('JAD HD:REAL DEAL:opener'!B13)
=MESSAGE(FALSE)
=ACTIVATE("Typeone")
access mode inputs if w value
=DIALOG.BOX('JAD HD:REAL DEAL:WACC1'!B3:H12)
=ECHO(FALSE)
=IF('JAD HD:REAL DEAL:WACC1'!$H$6=2)
=SET.NAME("walk",1)
=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:WACC1'!$I$8,'JAD
HD:REAL DEAL:WACC1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:WACC1'!$B$3:$H$12)
=NEXT()
=END.IF()
=IF('JAD HD:REAL DEAL:WACC1'!$H$6=3)
=SET.NAME("walk",1)
=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:WACC1'!$I$9,'JAD
HD:REAL DEAL:WACC1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:WACC1'!$B$3:$H$12)

```

```

=NEXT()
=END.IF()
=IF('JAD HD:REAL DEAL:WACC1'!$H$6=4)
=SET.NAME("walk",1)
=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:WACC1'!$I$10,'JAD
HD:REAL DEAL:WACC1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:WACC1'!$B$3:$H$12)
=NEXT()
=END.IF()
negative number loop =IF(OR('JAD HD:REAL DEAL:WACC1'!H$8<0,'JAD HD:REAL
DEAL:WACC1'!H10<0))
=ALERT("The input must be a positive number.",2)
=GOTO(Typeone!B114)
=END.IF()
print to summary sheet =MESSAGE(TRUE,"Recording the walk access inputs to the
mode summary statistics file.")
=ACTIVATE("WACC1")
=SELECT('JAD HD:REAL DEAL:WACC1'!H$5)
=COPY()
=ACTIVATE("Summary")
=SELECT(IB$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("WACC1")
rate =SELECT(!H$8)
=COPY()
=ACTIVATE("Summary")
=SELECT(ID$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("WACC1")
w value =SELECT(!H$10)
=COPY()
=ACTIVATE("Summary")
=SELECT(!F$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=MESSAGE(FALSE)
=ECHO(FALSE)
=ELSE()
=ACTIVATE("opener")
=MESSAGE(TRUE,"Opening the necessary files.")
open files;changes not saved =RUN('JAD HD:REAL DEAL:opener'!B19)
=ACTIVATE("Typeone")
=MESSAGE(FALSE)
inputs if given actual distance =DIALOG.BOX('JAD HD:REAL DEAL:ACCDIST1'!$B$3:$H$12)
=IF('JAD HD:REAL DEAL:ACCDIST1'!$H$6=2)
=SET.NAME("walk",1)

```

```

=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:ACCDIST1'!$I$8,'JAD
HD:REAL DEAL:ACCDIST1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:ACCDIST1'!$B$3:$H$12)
=NEXT()
=END.IF()
=IF('JAD HD:REAL DEAL:ACCDIST1'!$H$6=3)
=SET.NAME("walk",1)
=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:ACCDIST1'!$I$9,'JAD
HD:REAL DEAL:ACCDIST1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:ACCDIST1'!$B$3:$H$12)
=NEXT()
=END.IF()
=IF('JAD HD:REAL DEAL:ACCDIST1'!$H$6=4)
=SET.NAME("walk",1)
=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:ACCDIST1'!$I$10,'JAD
HD:REAL DEAL:ACCDIST1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:ACCDIST1'!$B$3:$H$12)
=NEXT()
=END.IF()
negative number loop
=IF(OR('JAD HD:REAL DEAL:ACCDIST1'!H$8<0,'JAD
HD:REAL DEAL:ACCDIST1'!H$10<0))
=ALERT("The input must be a positive number.",2)
=GOTO(Typeone!B172)
=END.IF()
print to summary sheet
mode
=MESSAGE(TRUE,"Recording the walk access inputs to the
summary statistics file.")
=ACTIVATE("ACCDIST1")
=SELECT('JAD HD:REAL DEAL:ACCDIST1'!H$5)
=COPY()
=ACTIVATE("Summary")
=SELECT(IB$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("ACCDIST1")
rate
=SELECT(!H$8)
=COPY()
=ACTIVATE("Summary")
=SELECT(!D$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("ACCDIST1")
actual distance
=SELECT(!H$10)
=COPY()
=ACTIVATE("Summary")

```

	=SELECT(!H\$6)
	=PASTE.SPECIAL(3,1)
<i>end walk access loop</i>	=ACTIVATE("Typeone")
	=END.IF()
	=MESSAGE(FALSE)
<i>form of wait calculation</i>	=DIALOG.BOX('JAD HD:REAL DEAL:wait'!\$B3:\$H9)
	=MESSAGE(TRUE,"Recording the waiting time inputs to the summary statistics file.")
<i>print to summary sheet</i>	=ACTIVATE("wait")
	=SELECT(H\$8)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(!F2)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
	=IF('JAD HD:REAL DEAL:wait'!H7=4)
<i>user specified wait time negative number loop</i>	=INPUT("Enter the waiting time in minutes.",2,"WAIT TIME")
	=IF(B233<0)
	=ALERT("The input must be a positive number.",2)
	=GOTO(Typeone!B233)
	=END.IF()
<i>print to summary sheet</i>	=SELECT(B233)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(!B\$10)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
	=END.IF()
	=MESSAGE(FALSE)
<i>use segments or not</i>	=INPUT("Do you wish to break the analysis into a series of segments along the route which will account for varying conditions (yes or no)?",2,"SEGMENTS","yes")
<i>if Poisson is used</i>	=INPUT("Do you wish to calculate the dwell time with Poisson probability theory (yes) or input a dwell time (no)?",2,"DWELL TIME","yes")
<i>no seg and Poisson calcs Poisson information negative number loop</i>	=IF(AND(B\$246="no",B\$247="yes"))
	=DIALOG.BOX('JAD HD:REAL DEAL:pax'!B\$3:H\$17)
	=IF(OR('JAD HD:REAL DEAL:pax'!H\$5<0,'JAD HD:REAL DEAL:pax'!H\$7<0,'JAD HD:REAL DEAL:pax'!H\$9<0,'JAD HD:REAL DEAL:pax'!H\$11<0,'JAD HD:REAL DEAL:pax'!H\$13,'JAD HD:REAL DEAL:pax'!H\$15<0))
	=ALERT("The input must be a positive number.",2)
	=GOTO(Typeone!B249)
	=END.IF()
	=MESSAGE(TRUE,"Recording the Poisson inputs to the summary statistics file.")
<i>print to summary sheet pax boarding demand</i>	=ACTIVATE("pax")
	=SELECT(!H\$5)
	=COPY()

	=ACTIVATE("Summary")
	=SELECT(IC\$11)
	=PASTE.SPECIAL(3,1)
<i>pax deboarding demand</i>	=ACTIVATE("pax")
	=SELECT(IH\$7)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(IE\$11)
	=PASTE.SPECIAL(3,1)
<i>pax headway</i>	=ACTIVATE("pax")
	=SELECT(IH\$9)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(IG\$11)
	=PASTE.SPECIAL(3,1)
<i>board start up time</i>	=ACTIVATE("pax")
	=SELECT(IH\$11)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(IC\$12)
	=PASTE.SPECIAL(3,1)
<i>deboard start up time</i>	=ACTIVATE("pax")
	=SELECT(IH\$13)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(IE\$12)
	=PASTE.SPECIAL(3,1)
<i>Poisson probability</i>	=ACTIVATE("pax")
	=SELECT(IH\$15)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(IG\$12)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
	=END.IF()
	=MESSAGE(FALSE)
<i>no seg and given dwell time</i>	=IF(AND(B\$246="no",B\$247="no"))
<i>given dwell time</i>	=INPUT("Enter the dwell time in seconds.",2,"DWELL TIME",45)
<i>negative number loop</i>	=IF(B295<0)
	=ALERT("The input must be a positive number.",2)
	=GOTO(Typeone!B295)
	=END.IF()
<i>print to summary sheet</i>	=SELECT(B295)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(ID\$10)
	=PASTE.SPECIAL(3,1)

<i>no segmentation</i>	=ACTIVATE("Typeone")
<i>initialize counter</i>	=END.IF()
<i>value of counter</i>	=IF(B246="no")
<i>loop to print incr if no seg</i>	=SET.NAME("solo",1)
	=solo
	=FOR("solo",1,20,1)
	=SELECT(Typeone!B9)
	=COPY()
	=SELECT(OFFSET(Typeone!G\$308,solo,0))
	=PASTE.SPECIAL(3,1)
	=NEXT()
	=GOTO(B\$490)
	=END.IF()
<i>if using segmentation</i>	=IF(B246="yes")
<i>number of segments</i>	=INPUT("Enter the number of segments.",1,"NUMBER OF SEGMENTS",3)
	=B319
<i>negative number loop</i>	=IF(B320<0)
	=ALERT("The input must be a positive number.",2)
	=GOTO(Typeone!B319)
	=END.IF()
	=ACTIVATE("Typeone")
	=COPY(!B302)
	=ACTIVATE("Summary")
	=SELECT(!D2)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
	=END.IF()
<i>seg and Poisson calcs</i>	=IF(AND(B246="yes",B247="yes"))
<i>message to explain cons&var</i>	=ALERT("Some of the required Poisson inputs are constant for each segment and the rest vary across the segments.",2)
	=DIALOG.BOX('JAD HD:REAL DEAL:pax'!B\$18:H\$28)
<i>negative number loop</i>	=IF(OR('JAD HD:REAL DEAL:pax'!H\$20<0,'JAD HD:REAL DEAL:pax'!H\$22<0,'JAD HD:REAL DEAL:pax'!H\$24<0,'JAD HD:REAL DEAL:pax'!H\$26<0))
	=ALERT("The input must be a positive number.",2)
	=GOTO(Typeone!B334)
	=END.IF()
	=MESSAGE(TRUE,"Recording the Poisson inputs to the summary statistics file.")
	=ACTIVATE("pax")
<i>pax headway</i>	=SELECT(!H\$20)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(!G\$11)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("pax")
<i>board start up time</i>	=SELECT(!H\$22)

deboard start up time

```
=COPY()  
=ACTIVATE("Summary")  
=SELECT(!C$12)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("pax")  
=SELECT(!H$24)  
=COPY()  
=ACTIVATE("Summary")  
=SELECT(!E$12)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("pax")  
=SELECT(!H$26)
```

Poisson probability

```
=COPY()  
=ACTIVATE("Summary")  
=SELECT(!G$12)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Typeone")  
=MESSAGE(FALSE)  
=ALERT("A dialog box will appear for each segment. Enter the  
corresponding segment variables in each successive dialog  
box.",2)  
=SET.NAME("length",1)  
=length
```

negative number loop

```
=FOR("length",1,B320,1)  
=DIALOG.BOX('JAD HD:REAL DEAL:variables!B$3:H$15)  
=IF(OR('JAD HD:REAL DEAL:variables!H$5<0,'JAD HD:REAL  
DEAL:variables!H$7<0,'JAD HD:REAL  
DEAL:variables!H$9<0,'JAD HD:REAL  
DEAL:variables!H$11<0,'JAD HD:REAL  
DEAL:variables!H$13<0))  
=ALERT("The input must be a positive number.",2)  
=GOTO(Typeone!B370)  
=END.IF()  
=MESSAGE(TRUE,"Recording the segment variables to the  
summary statistics file.")  
='JAD HD:REAL DEAL:variables!H$5  
=COPY(B376)  
=SELECT(OFFSET(Typeone!C$316,length,0))  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Summary")  
=SELECT(OFFSET('JAD HD:REAL  
DEAL:Summary!$C$12,length,0))  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Typeone")  
='JAD HD:REAL DEAL:variables!H$7  
=COPY(B384)  
=SELECT(OFFSET(Typeone!D$316,length,0))  
=PASTE.SPECIAL(3,1)
```

```

=ACTIVATE("Summary")
=SELECT(OFFSET('JAD HD:REAL
DEAL:Summary'!$E$12,length,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
='JAD HD:REAL DEAL:variables'!H$9
=COPY(B392)
=SELECT(OFFSET(Typeone!E$316,length,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Summary")
=SELECT(OFFSET('JAD HD:REAL
DEAL:Summary'!$G$12,length,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
='JAD HD:REAL DEAL:variables'!H$11
=COPY(B400)
=SELECT(OFFSET(Typeone!F$316,length,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Summary")
=SELECT(OFFSET('JAD HD:REAL
DEAL:Summary'!$I$12,length,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
='JAD HD:REAL DEAL:variables'!H$13
=COPY(B408)
=SELECT(OFFSET(Typeone!G$316,length,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Summary")
=SELECT(OFFSET('JAD HD:REAL
DEAL:Summary'!$K$12,length,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=MESSAGE(FALSE)
=NEXT()
=SUM(C$317:C$387)
segs too short
=IF(B418<'JAD HD:REAL DEAL:LHM1'!H$20)
=ALERT("The sum of the segment lengths is less than the route
length.",3)
=DIALOG.BOX('JAD HD:REAL DEAL:LHM1'!B$18:H$21)
=GOTO(B$369)
=END.IF()
segs too long
=IF(B418>'JAD HD:REAL DEAL:LHM1'!H$20)
=ALERT("The sum of the segment lengths is greater than the
route length.",3)
=DIALOG.BOX('JAD HD:REAL DEAL:LHM1'!B$18:H$21)
=GOTO(B$369)
=END.IF()
end seg and Poisson loop
=END.IF()

```

seg and given dwell time

```
=IF(AND(B$246="yes",B$247="no"))  
=ALERT("A dialog box will appear for each segment. Enter the  
corresponding segment variables in each successive dialog  
box.",2)
```

```
=SET.NAME("length",1)
```

```
=length
```

```
=FOR("length",1,B320,1)
```

```
=DIALOG.BOX('JAD HD:REAL DEAL:variables!B$16:H$24)
```

negative number loop

```
=IF(OR('JAD HD:REAL DEAL:variables!H$18<0,'JAD HD:REAL  
DEAL:variables!H$20<0,'JAD HD:REAL  
DEAL:variables!H$22<0))
```

```
=ALERT("The input must be a positive number.",2)
```

```
=GOTO(Typeone!B435)
```

```
=END.IF()
```

```
=MESSAGE(TRUE,"Recording the segment variables to the  
summary statistics file.")
```

```
=JAD HD:REAL DEAL:variables!H$18
```

```
=COPY(B441)
```

```
=SELECT(OFFSET(Typeone!C$316,length,0))
```

```
=PASTE.SPECIAL(3,1)
```

```
=ACTIVATE("Summary")
```

```
=SELECT(OFFSET(!C$12,length,0))
```

```
=PASTE.SPECIAL(3,1)
```

```
=ACTIVATE("Typeone")
```

```
=JAD HD:REAL DEAL:variables!H$20
```

```
=COPY(B449)
```

```
=SELECT(OFFSET(Typeone!F$316,length,0))
```

```
=PASTE.SPECIAL(3,1)
```

```
=ACTIVATE("Summary")
```

```
=SELECT(OFFSET(!I$12,length,0))
```

```
=PASTE.SPECIAL(3,1)
```

```
=ACTIVATE("Typeone")
```

```
=JAD HD:REAL DEAL:variables!H$22
```

```
=COPY(B457)
```

```
=SELECT(OFFSET(Typeone!G$316,length,0))
```

```
=PASTE.SPECIAL(3,1)
```

```
=ACTIVATE("Summary")
```

```
=SELECT(OFFSET(!K$12,length,0))
```

```
=PASTE.SPECIAL(3,1)
```

```
=ACTIVATE("Typeone")
```

```
=MESSAGE(FALSE)
```

```
=NEXT()
```

```
=SUM(C$317:C$387)
```

segs too short

```
=IF(B467<'JAD HD:REAL DEAL:LHM1!H$20)
```

```
=ALERT("The sum of the segment lengths is less than the route  
length.",3)
```

```
=DIALOG.BOX('JAD HD:REAL DEAL:LHM1!B$18:H$21)
```

```
=GOTO(B$434)
```

```

=END.IF()
segs too long =IF(B467>'JAD HD:REAL DEAL:LHM1'!H$20)
=ALERT("The sum of the segment lengths is greater than the
route length.",3)
=IALOG.BOX('JAD HD:REAL DEAL:LHM1'!B$18:H$21)
=GOTO(B$434)
=END.IF()
given dwell time =INPUT("Enter the dwell time in seconds.",2,"DWELL TIME",45)
negative number loop =IF(B478<0)
=ALERT("The input must be a positive number.",2)
=GOTO(Typeone!B478)
=END.IF()
print to summary sheet =SELECT(B478)
=COPY()
=ACTIVATE("Summary")
=SELECT(!D10)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=END.IF()
egress mode =IALOG.BOX('JAD HD:REAL DEAL:EGRESS1'!B3:H8)
print to summary sheet =ACTIVATE("EGRESS1")
=SELECT(!H$5)
=COPY()
=ACTIVATE("Summary")
=SELECT(!B$20)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=IF('JAD HD:REAL DEAL:WVAL1'!H6=1)
egress inputs if willing... =IF('JAD HD:REAL DEAL:EGRESS1'!H6=1)
begin walk loop =IALOG.BOX('JAD HD:REAL DEAL:WALKEGR'!B3:H9)
egress mode inputs if walk =IF(OR('JAD HD:REAL DEAL:WALKEGR'!H5<0,'JAD HD:REAL
DEAL:WALKEGR'!H7<0))
negative number loop =ALERT("The inputs must be positive numbers.",2)
=GOTO(Typeone!B500)
=END.IF()
=MESSAGE(TRUE,"Recording the walk egress inputs to the
summary statistics file.")
print to sumary sheet =ACTIVATE("WALKEGR")
=SELECT(!H$5)
=COPY()
=ACTIVATE("Summary")
=SELECT(!D$20)
walk egress rate =PASTE.SPECIAL(3,1)
=ACTIVATE("Walkegr")
=SELECT(!H$7)
=COPY()
=ACTIVATE("Summary")
w value =SELECT(!F$20)

```

```

end walk loop      =PASTE.SPECIAL(3,1)
                   =ACTIVATE("Typeone")
                   =END.IF()
begin bus/walk loop =MESSAGE(FALSE)
                   =IF('JAD HD:REAL DEAL:EGRESS1'!H6=2)
                   =MESSAGE(TRUE,"Recording the egress bus inputs to the
egress mode inputs if bus/walk summary statistics file.")
negative number loop =DIALOG.BOX('JAD HD:REAL DEAL:WBUSDIST1'!B3:H15)
                   =IF(OR('JAD HD:REAL DEAL:WBUSDIST1'!H$5<0,'JAD
HD:REAL DEAL:WBUSDIST1'!H$7<0,'JAD HD:REAL
DEAL:WBUSDIST1'!H$9<0))
                   =ALERT("The inputs must be positive numbers.",2)
                   =GOTO(Typeone!B523)
                   =END.IF()
                   =MESSAGE(TRUE,"Recording the bus egress inputs to the
summary statistics file.")
print to summary sheet =ACTIVATE("WBUSDIST1")
                       =SELECT(!H$5)
                       =COPY()
egress bus rate      =ACTIVATE("Summary")
                       =SELECT(!D$21)
                       =PASTE.SPECIAL(3,1)
                       =ACTIVATE("WBUSDIST1")
                       =SELECT(!H$7)
                       =COPY()
egress bus w value   =ACTIVATE("Summary")
                       =SELECT(!F$21)
                       =PASTE.SPECIAL(3,1)
                       =ACTIVATE("WBUSDIST1")
                       =SELECT(!H$9)
                       =COPY()
egress bus headway   =ACTIVATE("Summary")
                       =SELECT(!J$21)
                       =PASTE.SPECIAL(3,1)
                       =ACTIVATE("WBUSDIST1")
walk rate from egress bus =SELECT(!H$11)
                       =COPY()
                       =ACTIVATE("Summary")
                       =SELECT(!N$21)
                       =PASTE.SPECIAL(3,1)
                       =ACTIVATE("WBUSDIST1")
walk w value from egress bus =SELECT(!H$13)
                               =COPY()
                               =ACTIVATE("Summary")
                               =SELECT(!P$21)
                               =PASTE.SPECIAL(3,1)
                               =ACTIVATE("Typeone")
end bus/walk loop    =END.IF()

```

<i>begin combo loop</i>	=MESSAGE(FALSE)
<i>egress mode inputs if walk</i>	=IF('JAD HD:REAL DEAL:EGRESS1'!H6=3)
<i>negative number loop</i>	=DIALOG.BOX('JAD HD:REAL DEAL:WWALKEGR2'!B12:H20)
	=IF(OR('JAD HD:REAL DEAL:WWALKEGR2'!H\$14<0,'JAD HD:REAL DEAL:WWALKEGR2'!H\$16<0,'JAD HD:REAL DEAL:WWALKEGR2'!H\$18<0))
	=ALERT("The inputs must be positive numbers.",2)
	=GOTO(Typeone!B563)
	=END.IF()
<i>egress mode inputs if bus/walk</i>	=DIALOG.BOX('JAD HD:REAL DEAL:WBUSEGR2'!B3:H17)
<i>negative number loop</i>	=IF(OR('JAD HD:REAL DEAL:WBUSEGR2'!H\$5<0,'JAD HD:REAL DEAL:WBUSEGR2'!H\$7<0,'JAD HD:REAL DEAL:WBUSEGR2'!H\$9<0,'JAD HD:REAL DEAL:WBUSEGR2'!H\$11<0))
	=ALERT("The inputs must be positive numbers.",2)
	=GOTO(Typeone!B568)
	=END.IF()
	=IF(OR('JAD HD:REAL DEAL:WWALKEGR2'!H14+'JAD HD:REAL DEAL:WBUSEGR2'!H5>100,'JAD HD:REAL DEAL:WWALKEGR2'!H14+'JAD HD:REAL DEAL:WBUSEGR2'!H5<100))
<i>check that % add to 100</i>	=ALERT("Egress mode percentages must add to 100%.",2)
	=GOTO(B563)
	=END.IF()
	=MESSAGE(TRUE,"Recording the combination egress inputs to the summary statistics file.")
	=ACTIVATE("WWALKEGR2")
	=SELECT(!H\$14)
	=COPY()
<i>walk egress percent</i>	=ACTIVATE("Summary")
	=SELECT(!L\$20)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("WWALKEGR2")
	=SELECT(!H\$16)
	=COPY()
	=ACTIVATE("Summary")
<i>walk egress rate</i>	=SELECT(!D\$20)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("WWALKEGR2")
	=SELECT(!H\$18)
	=COPY()
	=ACTIVATE("Summary")
<i>walk egress w value</i>	=SELECT(!F\$20)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("WBUSEGR2")
	=SELECT(!H\$5)
	=COPY()
	=ACTIVATE("Summary")
<i>bus egress percent</i>	=SELECT(!L\$21)


```

= PASTE.SPECIAL(3,1)
= ACTIVATE("WBUSEGR2")
= SELECT(!H$7)
= COPY()
bus egress rate
= ACTIVATE("Summary")
= SELECT(!D$21)
= PASTE.SPECIAL(3,1)
= ACTIVATE("WBUSEGR2")
= SELECT(!H$9)
= COPY()
bus egress w value
= ACTIVATE("Summary")
= SELECT(!F$21)
= PASTE.SPECIAL(3,1)
= ACTIVATE("WBUSEGR2")
= SELECT(!H$11)
= COPY()
bus egress headway
= ACTIVATE("Summary")
= SELECT(!J$21)
= PASTE.SPECIAL(3,1)
= ACTIVATE("WBUSEGR2")
= SELECT(!H$13)
= COPY()
walk rate from egress bus
= ACTIVATE("Summary")
= SELECT(!N$21)
= PASTE.SPECIAL(3,1)
= ACTIVATE("WBUSEGR2")
= SELECT(!H$15)
= COPY()
walk w value from egress bus
= ACTIVATE("Summary")
= SELECT(!P$21)
= PASTE.SPECIAL(3,1)
= ACTIVATE("Typeone")
end combo loop
= END.IF()
= MESSAGE(FALSE)
= ELSE()
switch to actual egress dist
= IF('JAD HD:REAL DEAL:EGRESS1'!H6=1)
begin walk loop
= DIALOG.BOX('JAD HD:REAL DEAL:WALKEGR'!B10:H16)
walk egress inputs
= IF(OR('JAD HD:REAL DEAL:WALKEGR'!H$12<0,'JAD
HD:REAL DEAL:WALKEGR'!H$14<0))
negative number loop
= ALERT("The inputs must be positive numbers.",2)
= GOTO(Typeone!B637)
= END.IF()
= MESSAGE(TRUE,"Recording the walk egress inputs to the
summary statistics file.")
print to summary sheet
= ACTIVATE("WALKEGR")
= SELECT(!H$12)
= COPY()

```

<i>walk egress rate</i>	=ACTIVATE("Summary") =SELECT(ID\$20) =PASTE.SPECIAL(3,1) =ACTIVATE("Walkegr") =SELECT(!H\$14) =COPY()
<i>walk distance</i>	=ACTIVATE("Summary") =SELECT(!H\$20) =PASTE.SPECIAL(3,1) =ACTIVATE("Typeone")
<i>end walk loop</i>	=END.IF() =MESSAGE(FALSE)
<i>begin bus/walk loop</i>	=IF('JAD HD:REAL DEAL:EGRESS1'!H6=2)
<i>bus/walk egress inputs</i>	=DIALOG.BOX('JAD HD:REAL DEAL:BusEgr1'!B3:H15)
<i>negative number loop</i>	=IF(OR('JAD HD:REAL DEAL:BusEgr1'!H\$5<0,'JAD HD:REAL DEAL:BusEgr1'!H\$7<0,'JAD HD:REAL DEAL:BusEgr1'!H\$9<0)) =ALERT("The inputs must be positive numbers.",2) =GOTO(Typeone!B659) =END.IF()
<i>print to summary sheet</i>	=MESSAGE(TRUE,"Recording the bus egress inputs to the summary statistics file.") =ACTIVATE("BUSEGR1") =SELECT(!H\$5) =COPY()
<i>egress bus rate</i>	=ACTIVATE("Summary") =SELECT(ID\$21) =PASTE.SPECIAL(3,1) =ACTIVATE("BUSEGR1") =SELECT(!H\$7) =COPY()
<i>egress bus distance</i>	=ACTIVATE("Summary") =SELECT(!H\$21) =PASTE.SPECIAL(3,1) =ACTIVATE("BUSEGR1") =SELECT(!H\$9) =COPY()
<i>egress bus headway</i>	=ACTIVATE("Summary") =SELECT(!J\$21) =PASTE.SPECIAL(3,1) =ACTIVATE("BUSEGR1") =SELECT(!H\$11) =COPY()
<i>walk rate from egress bus</i>	=ACTIVATE("Summary") =SELECT(!N\$21) =PASTE.SPECIAL(3,1) =ACTIVATE("BUSEGR1")

```

=SELECT(!H$13)
=COPY()
=ACTIVATE("Summary")
walk dist from egress bus
=SELECT(!R$21)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=MESSAGE(FALSE)
end bus/walk loop
begin combo input loop
combo inputs
negative number loop
=END.IF()
=IF('JAD HD:REAL DEAL:EGRESS1'!H6=3)
=IALOG.BOX('JAD HD:REAL DEAL:WalkEgr2'!B12:H20)
=IF(OR('JAD HD:REAL DEAL:WalkEgr2'!H$14<0,'JAD
HD:REAL DEAL:WalkEgr2'!H$16<0,'JAD HD:REAL
DEAL:WalkEgr2'!H$18<0))
=ALERT("The inputs must be positive numbers.",2)
=GOTO(Typeone!B699)
=END.IF()
=IALOG.BOX('JAD HD:REAL DEAL:BusEgr2'!B3:H17)
negative number loop
=IF(OR('JAD HD:REAL DEAL:BusEgr2'!H$5<0,'JAD HD:REAL
DEAL:BusEgr2'!H$7<0,'JAD HD:REAL
DEAL:BusEgr2'!H$9<0,'JAD HD:REAL
DEAL:BusEgr2'!H$11<0))
=ALERT("The inputs must be positive numbers.",2)
=GOTO(Typeone!B704)
=END.IF()
=IF(OR('JAD HD:REAL DEAL:WalkEgr2'!H14+'JAD HD:REAL
DEAL:BusEgr2'!H5>100,'JAD HD:REAL
DEAL:WWALKEGR2'!H14+'JAD HD:REAL
DEAL:WBUSEGR2'!H5<100))
check that % add to 100
=ALERT("Egress mode percentages must add to 100%.",2)
=GOTO(B699)
=END.IF()
=MESSAGE(TRUE,"Recording the egress combination inputs
to the summary statistics file.")
=ACTIVATE("WALKEGR2")
=SELECT(!H$14)
=COPY()
=ACTIVATE("Summary")
walk egress percent
=SELECT(!L$20)
=PASTE.SPECIAL(3,1)
=ACTIVATE("WALKEGR2")
=SELECT(!H$16)
=COPY()
=ACTIVATE("Summary")
walk egress rate
=SELECT(!D$20)
=PASTE.SPECIAL(3,1)
=ACTIVATE("WALKEGR2")
=SELECT(!H$18)
=COPY()
=ACTIVATE("Summary")

```

<i>walk egress distance</i>	=SELECT(!H\$20) =PASTE.SPECIAL(3,1) =ACTIVATE("BusEgr2") =SELECT(!H\$5) =COPY() =ACTIVATE("Summary")
<i>bus egress percent</i>	=SELECT(!L\$21) =PASTE.SPECIAL(3,1) =ACTIVATE("BusEgr2") =SELECT(!H\$7) =COPY() =ACTIVATE("Summary")
<i>bus egress rate</i>	=SELECT(!D\$21) =PASTE.SPECIAL(3,1) =ACTIVATE("BusEgr2") =SELECT(!H\$9) =COPY() =ACTIVATE("Summary")
<i>bus egress distance</i>	=SELECT(!F\$21) =PASTE.SPECIAL(3,1) =ACTIVATE("BusEgr2") =SELECT(!H\$11) =COPY() =ACTIVATE("Summary")
<i>bus egress headway</i>	=SELECT(!J\$21) =PASTE.SPECIAL(3,1) =ACTIVATE("BusEgr2") =SELECT(!H\$13) =COPY() =ACTIVATE("Summary")
<i>walk rate from egress bus</i>	=SELECT(!N\$21) =PASTE.SPECIAL(3,1) =ACTIVATE("BusEgr2") =SELECT(!H\$15) =COPY() =ACTIVATE("Summary")
<i>walk dist from egress bus</i>	=SELECT(!R\$21) =PASTE.SPECIAL(3,1) =ACTIVATE("Typeone")
<i>end combo input loop</i>	=END.IF()
<i>end loop for egress inputs</i>	=END.IF() =MESSAGE(FALSE) =ECHO(FALSE)
<i>counter for printing columns</i>	=SET.NAME("answer",0) =SET.NAME("seg",1)
<i>skip counter stuff for solo</i>	=IF(B246="no") =GOTO(B\$787)

	=END.IF()
	=WHILE(OR(seg<B320,seg=B320))
	=MESSAGE(FALSE)
	=answer
	=seg
	=COPY(B781)
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL
	DEAL:WORKSHEETS'!\$A\$3,0,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
<i>initialize counter to zero</i>	=SET.NAME("count",0)
	=OFFSET(Typeone!G\$316,seg,0)
<i>test conditions to continue loop</i>	=
	WHILE(OR(B20+count*B788<B31,B20+count*B788
	=B31))
<i>display the spacing</i>	=B20+(B788*count)
	=MESSAGE(TRUE,"Computing the waiting time costs.")
<i>wait time loop</i>	=IF('JAD HD:REAL DEAL:wait'!\$H\$7=1)
<i>half headway (wait)</i>	='JAD HD:REAL DEAL:LHM1'!H9/2
	=END.IF()
	=IF('JAD HD:REAL DEAL:wait'!\$H\$7=2)
<i>sq root of headway (wait)</i>	='JAD HD:REAL DEAL:LHM1'!H9^0.5
	=END.IF()
	=IF('JAD HD:REAL DEAL:wait'!\$H\$7=3)
<i>30 min criteria</i>	=IF(OR('JAD HD:REAL DEAL:LHM1'!H9>30,'JAD HD:REAL
	DEAL:LHM1'!H9=30))
	='JAD HD:REAL DEAL:LHM1'!H9/2
	=ELSE()
	='JAD HD:REAL DEAL:LHM1'!H9^0.5
	=END.IF()
	=END.IF()
	=MESSAGE(FALSE)
	=MESSAGE(TRUE,"Computing the access and egress costs.")
<i>if stmt if willing to travel</i>	=IF('JAD HD:REAL DEAL:WVAL1'!H6=1)
<i>walk distance</i>	=(('JAD HD:REAL DEAL:WACC1'!H10^2)+(('JAD HD:REAL
	DEAL:LHM1'!H7/2)^2)^0.5
<i>walk time min</i>	=B808/('JAD HD:REAL DEAL:WACC1'!H8*60)
<i>begin walk egress calcs</i>	=IF('JAD HD:REAL DEAL:EGRESS1'!H6=1)
<i>walk egress distance</i>	=(('JAD HD:REAL DEAL:WALKEGR1'!H7^2)+(('JAD HD:REAL
	DEAL:LHM1'!H7/2)^2)^0.5
<i>walk egress time min</i>	=B811/('JAD HD:REAL DEAL:WALKEGR1'!H5*60)
<i>acc/egr time if walk</i>	=B809+B812
<i>end walk calculations</i>	=END.IF()
<i>begin bus/walk calculations</i>	=IF('JAD HD:REAL DEAL:EGRESS1'!H6=2)

<i>bus/walk egress</i>	$=('JAD\ HD:REAL\ DEAL:WBUSDIST1'H7)/('JAD\ HD:REAL\ DEAL:WBUSDIST1'H5/60))+(((('JAD\ HD:REAL\ DEAL:WBUSDIST1'H13^2)+('JAD\ HD:REAL\ DEAL:LHM1'H7/2)^2)^0.5)/('JAD\ HD:REAL\ DEAL:WBUSDIST1'H11*60))$
<i>wait time for egress bus</i>	$=JAD\ HD:REAL\ DEAL:WBUSDIST1'H9^0.5$
<i>bus egress total time</i>	$=B816+B817$
<i>acc/egr time if bus/walk</i>	$=B809+B818$
<i>end if loop</i>	$=END.IF()$
<i>begin combo loop</i>	$=IF('JAD\ HD:REAL\ DEAL:EGRESS1'H6=3)$
<i>walk egress distance</i>	$=(('JAD\ HD:REAL\ DEAL:WWALKEGR2'H18^2)+('JAD\ HD:REAL\ DEAL:LHM1'H7/2)^2)^0.5$
<i>walk egress time min</i>	$=B822/('JAD\ HD:REAL\ DEAL:WWALKEGR2'H16*60)$
<i>acc/egr time if walk</i>	$=B809+B823$
<i>bus/walk egress</i>	$=(('JAD\ HD:REAL\ DEAL:WBUSEGR2'H9)/('JAD\ HD:REAL\ DEAL:WBUSEGR2'H7/60))+(((('JAD\ HD:REAL\ DEAL:WBUSEGR2'H15^2)+('JAD\ HD:REAL\ DEAL:LHM1'H7/2)^2)^0.5)/('JAD\ HD:REAL\ DEAL:WBUSEGR2'H13*60))$
<i>wait time for egress bus</i>	$=JAD\ HD:REAL\ DEAL:WBUSEGR2'H11^0.5$
<i>combo total egress time</i>	$=B823+B825+B826$
<i>acc/egr time if bus/walk</i>	$=B809+B825+B826$
<i>weighted time</i>	$=(('JAD\ HD:REAL\ DEAL:WWALKEGR2'H14/100)*(Typeone!B824))+('JAD\ HD:REAL\ DEAL:WBUSEGR2'H5/100)*(Typeone!B828)$
<i>end combo loop</i>	$=END.IF()$
<i>if providing actual distance</i>	$=ELSE()$
<i>access time</i>	$=('JAD\ HD:REAL\ DEAL:ACCDIST1'H10/('JAD\ HD:REAL\ DEAL:ACCDIST1'H8*60))$
<i>begin if loop</i>	$=IF('JAD\ HD:REAL\ DEAL:EGRESS1'H6=1)$
<i>walk egress time</i>	$=JAD\ HD:REAL\ DEAL:WWALKEGR1'H7/('JAD\ HD:REAL\ DEAL:WWALKEGR1'H5*60)$
<i>acc/egr time if walk</i>	$=B832+B834$
<i>end walk egress loop</i>	$=END.IF()$
<i>begin bus/walk loop</i>	$=IF('JAD\ HD:REAL\ DEAL:EGRESS1'H6=2)$
<i>bus egress time</i>	$=('JAD\ HD:REAL\ DEAL:BusEgr1'H7/('JAD\ HD:REAL\ DEAL:BusEgr1'H5/60))+(((('JAD\ HD:REAL\ DEAL:BusEgr1'H13^2)+('JAD\ HD:REAL\ DEAL:LHM1'H7/2)^2)^0.5)/('JAD\ HD:REAL\ DEAL:BusEgr1'H11*60))$
<i>wait time for egress bus</i>	$=JAD\ HD:REAL\ DEAL:BusEgr1'H9^0.5$
<i>bus egress total time</i>	$=B838+B839$
<i>acc/egr time if bus/walk</i>	$=B832+B840$
<i>end bus/walkloop</i>	$=END.IF()$
<i>begin combo loop</i>	$=IF('JAD\ HD:REAL\ DEAL:EGRESS1'H6=3)$
<i>walk egress min</i>	$=('JAD\ HD:REAL\ DEAL:WalkEgr2'H18/('JAD\ HD:REAL\ DEAL:WalkEgr2'H16*60))$
<i>acc/egr walk dist time</i>	$=B832+B844$

```

bus/walk egress min      =(('JAD HD:REAL DEAL:BusEgr2'!H9/('JAD HD:REAL
                        DEAL:BusEgr2'!H7/60))+(((('JAD HD:REAL
                        DEAL:BusEgr2'!H15^2))+((('JAD HD:REAL
                        DEAL:LHM1'!H7/2)^2))^0.5)/('JAD HD:REAL
                        DEAL:BusEgr2'!H13^60))
egress bus wait time    =JAD HD:REAL DEAL:BusEgr2'!H11^0.5
acc/egr time if bus/walk =B832+B846+B847
combo egress total time =B844+B846+B847
weighted distance time  =(('JAD HD:REAL
                        DEAL:WalkEgr2'!H14/100)*Typeone!B845)+((('JAD HD:REAL
                        DEAL:BusEgr2'!H5/100)*Typeone!B849)
end combo loop          =END.IF()
end if actual distance loop =END.IF()
                        =MESSAGE(TRUE,"Recording the access costs to the results
                        file.")
first segment           =IF(B246="no")
                        =ACTIVATE("Typeone")
                        =SELECT(B790)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
print the spacing on screen =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!A$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typeone")
access w value print loop =IF('JAD HD:REAL DEAL:WVAL1'!H$6=1)
                        =SELECT(B809)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!B$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typeone")
actual distance loop    =ELSE()
print access walk distance =SELECT(B832)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!B$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typeone")
end walk print loop     =END.IF()
                        =MESSAGE(TRUE,"Recording the waiting time costs to the
                        results file.")
half headway            =IF('JAD HD:REAL DEAL:wait'!H$7=1)
print wait time         =SELECT(B793)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!C$3,count,0))
                        =PASTE.SPECIAL(3,1)

```

```

=ACTIVATE("Typeone")
=END.IF()
sq root headway
print wait time
=IF('JAD HD:REAL DEAL:wait'!$H$7=2)
=SELECT(B796)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET'!C$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=END.IF()
30 min criteria
=IF('JAD HD:REAL DEAL:wait'!$H$7=3)
=IF(OR('JAD HD:REAL DEAL:LHM1'!H$9>30,'JAD HD:REAL
DEAL:LHM1'!H$9=30))
print wait time
=SELECT(B800)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET'!C$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=ELSE()
print wait time
=SELECT(B802)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET'!C$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=END.IF()
=END.IF()
user specified time
print wait time
=IF('JAD HD:REAL DEAL:wait'!$H$7=4)
=SELECT(B233)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET'!C$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typeone")
=END.IF()
=MESSAGE(TRUE,"Recording the egress costs to the results
file.")
walk egress print loop
=IF('JAD HD:REAL DEAL:EGRESS1'!H$6=1)
=IF('JAD HD:REAL DEAL:WVAL1'!H$6=1)
print walk w value
=SELECT(B812)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET'!F$3,count,0))

```


	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
	=ELSE()
<i>print walk distance</i>	=SELECT(B834)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
<i>end w value or distance loop</i>	=END.IF()
<i>end walk loop</i>	=END.IF()
<i>bus/walk egress print loop</i>	=IF('JAD HD:REAL DEAL:EGRESS1'!H\$6=2)
	=IF('JAD HD:REAL DEAL:WVAL1'!H\$6=1)
<i>print bus/walk w value</i>	=SELECT(B818)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ELSE()
<i>print bus/walk distance</i>	=SELECT(B840)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
<i>end w value or distance loop</i>	=END.IF()
<i>end bus/walk loop</i>	=END.IF()
<i>combo egress print loop</i>	=IF('JAD HD:REAL DEAL:EGRESS1'!H\$6=3)
	=IF('JAD HD:REAL DEAL:WVAL1'!H\$6=1)
<i>print combo w value</i>	=SELECT(B829)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ELSE()
<i>print combo distance</i>	=SELECT(B850)
	=COPY()
	=ACTIVATE("WORKSHEET")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEET'!F\$3,count,0))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
<i>end w value or distance loop</i>	=END.IF()
<i>end combo loop</i>	=END.IF()

<i>end no segment print loop</i>	=END.IF()
<i>print segments</i>	=MESSAGE(FALSE)
<i>show end product</i>	=IF(B246="yes")
	=ECHO(FALSE)
	=MESSAGE(TRUE,"Recording the access costs to the results file.")
	=ACTIVATE("Typeone")
	=SELECT(B790)
	=COPY()
	=ACTIVATE("WORKSHEETS")
<i>print the spacing on screen</i>	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!B\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
<i>access w value print loop</i>	=IF('JAD HD:REAL DEAL:WVAL1'!H\$6=1)
	=SELECT(B809)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
<i>switch to actual distance</i>	=ELSE()
<i>print access walk distance</i>	=SELECT(B832)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!C\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
<i>end w value or distance loop</i>	=END.IF()
	=MESSAGE(TRUE,"Recording the waiting time costs to the results file.")
<i>half headway</i>	=IF('JAD HD:REAL DEAL:wait'!\$H\$7=1)
<i>print wait time</i>	=SELECT(B793)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!D\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
	=END.IF()
<i>sq root headway</i>	=IF('JAD HD:REAL DEAL:wait'!\$H\$7=2)
<i>print wait time</i>	=SELECT(B796)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!D\$3,count,answer))
	=PASTE.SPECIAL(3,1)

	=ACTIVATE("Typeone")
	=END.IF()
<i>30 min criteria</i>	=IF('JAD HD:REAL DEAL:wait'!\$H\$7=3)
	=IF(OR('JAD HD:REAL DEAL:LHM1'!H\$9>30,'JAD HD:REAL DEAL:LHM1'!H\$9=30))
<i>print wait time</i>	=SELECT(B800)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!D\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
	=ELSE()
<i>print wait time</i>	=SELECT(B802)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!D\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
	=END.IF()
	=END.IF()
<i>user specified time</i>	=IF('JAD HD:REAL DEAL:wait'!\$H\$7=4)
<i>print wait time</i>	=SELECT(B233)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!D\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
	=END.IF()
<i>end input wait time loop</i>	=IF('JAD HD:REAL DEAL:EGRESS1'!H\$6=1)
<i>walk egress print loop</i>	=IF('JAD HD:REAL DEAL:WVAL1'!H\$6=1)
<i>print walk w value</i>	=SELECT(B812)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")
	=ELSE()
<i>print walk distance</i>	=SELECT(B834)
	=COPY()
	=ACTIVATE("WORKSHEETS")
	=SELECT(OFFSET('JAD HD:REAL DEAL:WORKSHEETS'!F\$3,count,answer))
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("Typeone")

```

end w value or distance loop      =END.IF()
end bus/walk loop                 =END.IF()
                                  =MESSAGE(TRUE,"Recording the egress costs to the results
                                  file.")
bus/walk egress print loop        =IF('JAD HD:REAL DEAL:EGRESS1'!H$6=2)
                                  =IF('JAD HD:REAL DEAL:WVAL1'!H$6=1)
print bus/walk w value            =SELECT(B818)
                                  =COPY()
                                  =ACTIVATE("WORKSHEETS")
                                  =SELECT(OFFSET('JAD HD:REAL
                                  DEAL:WORKSHEETS'!F$3,count,answer))
                                  =PASTE.SPECIAL(3,1)
                                  =ELSE()
print bus/walk distance           =SELECT(B840)
                                  =COPY()
                                  =ACTIVATE("WORKSHEETS")
                                  =SELECT(OFFSET('JAD HD:REAL
                                  DEAL:WORKSHEETS'!F$3,count,answer))
                                  =PASTE.SPECIAL(3,1)
                                  =ACTIVATE("Typeone")
end w value or distance loop      =END.IF()
end bus/walk loop                 =END.IF()
combo egress print loop           =IF('JAD HD:REAL DEAL:EGRESS1'!H$6=3)
                                  =IF('JAD HD:REAL DEAL:WVAL1'!H$6=1)
print combo w value               =SELECT(B829)
                                  =COPY()
                                  =ACTIVATE("WORKSHEETS")
                                  =SELECT(OFFSET('JAD HD:REAL
                                  DEAL:WORKSHEETS'!F$3,count,answer))
                                  =PASTE.SPECIAL(3,1)
                                  =ELSE()
print combo distance              =SELECT(B850)
                                  =COPY()
                                  =ACTIVATE("WORKSHEETS")
                                  =SELECT(OFFSET('JAD HD:REAL
                                  DEAL:WORKSHEETS'!F$3,count,answer))
                                  =PASTE.SPECIAL(3,1)
                                  =ACTIVATE("Typeone")
end w value or distance loop      =END.IF()
end combo loop                    =END.IF()
end second segment print         =END.IF()
                                  =MESSAGE(FALSE)
                                  =ECHO(FALSE)
                                  =MESSAGE(TRUE,"Incrementing the loop counter.")
increment counter                 =SET.NAME("count",count+1)
return to test condition          = NEXT()
                                  =ECHO(FALSE)
                                  =ACTIVATE("Linehaul")

```

turn control to linehaul

*don't incr counter if one seg
clear zeroes from results*

*increment answer counter
increment seg counter
return to multiple segments*

clear zeroes from results

return control to backbone

```
=MESSAGE(TRUE,"Computing the in-vehicle costs.")
=RUN('JAD HD:REAL DEAL:Linehaul'!B2)
=ACTIVATE("Typeone")
=MESSAGE(FALSE)
=IF(B246="no")
=ACTIVATE("WORKSHEET")
=SELECT('JAD HD:REAL DEAL:WORKSHEET'!$G$3:$I$25)
=FOR.CELL("CURRENT",,TRUE)
= CURRENT
= IF(CURRENT=0)
= FORMULA(" ",CURRENT)
= END.IF()
=NEXT()
=GOTO(B$1124)
=END.IF()
=MESSAGE(TRUE,"Incrementing the loop counters.")
=SET.NAME("answer",answer+8)
=SET.NAME("seg",seg+1)
=NEXT()
=MESSAGE(FALSE)
=ACTIVATE("WORKSHEETS")
=SELECT('JAD HD:REAL DEAL:WORKSHEETS'!G3:AV25)
=FOR.CELL("CURRENT",,TRUE)
= CURRENT
= IF(CURRENT=0)
= FORMULA(" ",CURRENT)
= END.IF()
=NEXT()
=ACTIVATE("Typeone")
=MESSAGE(FALSE)
=RETURN()
```


Typetwo Subroutine

<i>name</i>	Typetwo =ECHO(FALSE) =MESSAGE(TRUE,"Opening the necessary files.")
<i>open opener2</i>	=OPEN("JAD HD:REAL DEAL:opener2")
<i>open files for whole macro</i>	=RUN(opener2!B2)
<i>return control to this macro</i>	=ACTIVATE("Typetwo") =MESSAGE(FALSE)
<i>name to save results under</i>	=INPUT("Enter a name under which to save the results from this run.",2,"SAVE RESULTS")
<i>increment for loop</i>	=INPUT("Enter the spacing increment (ft):",1,"SPACING INCREMENT",400) =IF(OR(B9<0,B9=0))
<i>incorrect number entered</i>	=ALERT("The increment must be a number greater than 0.",3) =GOTO(B9) =END.IF() =SELECT(B\$9) =COPY() =ACTIVATE("Summary") =SELECT('JAD HD:REAL DEAL:Summary!C\$4)
<i>print the spacing on screen</i>	=PASTE.SPECIAL(3,1) =ACTIVATE("Typetwo")
<i>minimum spacing</i>	=INPUT("Enter the minumum spacing (ft):",1,"MINIMUM SPACING",400) =IF(OR(B20<0,B20=0))
<i>incorrect number entered</i>	=ALERT("The spacing must be a number greater than 0.",3) =GOTO(B20) =END.IF() =SELECT(B\$20) =COPY() =ACTIVATE("Summary") =SELECT('JAD HD:REAL DEAL:Summary!E\$4)
<i>print the spacing on screen</i>	=PASTE.SPECIAL(3,1) =ACTIVATE("Typetwo")
<i>maximum spacing</i>	=INPUT("Enter the maximum spacing (ft):",1,"MAXIMUM SPACING",6000) =IF(OR(B31<0,B31=0))
<i>incorrect number entered</i>	=ALERT("The spacing must be a number greater than 0.",3) =GOTO(B31) =END.IF() =SELECT(B\$31) =COPY() =ACTIVATE("Summary") =SELECT('JAD HD:REAL DEAL:Summary!G\$4)
<i>print the spacing on screen</i>	=PASTE.SPECIAL(3,1) =ACTIVATE("Typetwo")

linehaul inputs

```
=ECHO(FALSE)
=DIALOG.BOX('JAD HD:REAL DEAL:LHM2'!B3:H21)
=IF('JAD HD:REAL DEAL:LHM2'!H6=2)
=SET.NAME("display",1)
=WHILE(display=1)
=FORMULA('JAD HD:REAL DEAL:LHM2'!I8,'JAD HD:REAL
DEAL:LHM2'!H8)
=FORMULA('JAD HD:REAL DEAL:LHM2'!I10,'JAD HD:REAL
DEAL:LHM2'!H10)
=FORMULA('JAD HD:REAL DEAL:LHM2'!I12,'JAD HD:REAL
DEAL:LHM2'!H12)
=FORMULA('JAD HD:REAL DEAL:LHM2'!I14,'JAD HD:REAL
DEAL:LHM2'!H14)
=FORMULA('JAD HD:REAL DEAL:LHM2'!I16,'JAD HD:REAL
DEAL:LHM2'!H16)
=FORMULA('JAD HD:REAL DEAL:LHM2'!I19,'JAD HD:REAL
DEAL:LHM2'!H19)
=SET.NAME("display",display+1)
=DIALOG.BOX('JAD HD:REAL DEAL:LHM2'!B3:H21)
=NEXT()
=END.IF()
=IF('JAD HD:REAL DEAL:LHM2'!H6=3)
=SET.NAME("display",1)
=WHILE(display=1)
=FORMULA('JAD HD:REAL DEAL:LHM2'!J8,'JAD HD:REAL
DEAL:LHM2'!H8)
=FORMULA('JAD HD:REAL DEAL:LHM2'!J10,'JAD HD:REAL
DEAL:LHM2'!H10)
=FORMULA('JAD HD:REAL DEAL:LHM2'!J12,'JAD HD:REAL
DEAL:LHM2'!H12)
=FORMULA('JAD HD:REAL DEAL:LHM2'!J14,'JAD HD:REAL
DEAL:LHM2'!H14)
=FORMULA('JAD HD:REAL DEAL:LHM2'!J16,'JAD HD:REAL
DEAL:LHM2'!H16)
=FORMULA('JAD HD:REAL DEAL:LHM2'!J19,'JAD HD:REAL
DEAL:LHM2'!H19)
=SET.NAME("display",display+1)
=DIALOG.BOX('JAD HD:REAL DEAL:LHM2'!B3:H21)
=NEXT()
=END.IF()
=IF('JAD HD:REAL DEAL:LHM2'!H6=4)
=SET.NAME("display",1)
=WHILE(display=1)
=FORMULA('JAD HD:REAL DEAL:LHM2'!K8,'JAD HD:REAL
DEAL:LHM2'!H8)
=FORMULA('JAD HD:REAL DEAL:LHM2'!K10,'JAD HD:REAL
DEAL:LHM2'!H10)
=FORMULA('JAD HD:REAL DEAL:LHM2'!K12,'JAD HD:REAL
DEAL:LHM2'!H12)
=FORMULA('JAD HD:REAL DEAL:LHM2'!K14,'JAD HD:REAL
DEAL:LHM2'!H14)
```



```

=FORMULA('JAD HD:REAL DEAL:LHM2'!K16,'JAD HD:REAL
DEAL:LHM2'!H16)
=FORMULA('JAD HD:REAL DEAL:LHM2'!K19,'JAD HD:REAL
DEAL:LHM2'!H19)
=SET.NAME("display",display+1)
=DIALOG.BOX('JAD HD:REAL DEAL:LHM2'!B3:H21)
=NEXT()
=END.IF()
=IF(OR('JAD HD:REAL DEAL:LHM2'!H8<0,'JAD HD:REAL
DEAL:LHM2'!H10<0,'JAD HD:REAL DEAL:LHM2'!H12<0,'JAD
HD:REAL DEAL:LHM2'!H14<0,'JAD HD:REAL
DEAL:LHM2'!H16<0,'JAD HD:REAL DEAL:LHM2'!H19<0))
=ALERT("The inputs must be positive numbers.",2)
=GOTO(Typetwo!B43)
=END.IF()
=ECHO(FALSE)
=MESSAGE(TRUE,"Recording the Linehaul inputs to the
summary statistics file.")
=ACTIVATE("LHM2")
=SELECT('JAD HD:REAL DEAL:LHM2'!H$5)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary'!B$3)
print line haul mode
=PASTE.SPECIAL(3,1)
=ACTIVATE("LHM2")
=SELECT('JAD HD:REAL DEAL:LHM2'!H$8)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary'!F$3)
print route headway
=PASTE.SPECIAL(3,1)
=ACTIVATE("LHM2")
=SELECT('JAD HD:REAL DEAL:LHM2'!H$10)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary'!F$10)
print acceleration rate
=PASTE.SPECIAL(3,1)
=ACTIVATE("LHM2")
=SELECT('JAD HD:REAL DEAL:LHM2'!H$12)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary'!H$10)
print deceleration rate
=PASTE.SPECIAL(3,1)
=ACTIVATE("LHM2")
=SELECT('JAD HD:REAL DEAL:LHM2'!H$14)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary'!J$10)
print cruise velocity
=PASTE.SPECIAL(3,1)

```

```

=ACTIVATE("LHM2")
=SELECT('JAD HD:REAL DEAL:LHM2!H$16)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary!H$3)
print average trip length =PASTE.SPECIAL(3,1)
=ACTIVATE("LHM2")
=SELECT('JAD HD:REAL DEAL:LHM2!H$19)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary!J$3)
print route length =PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
w value preference =DIALOG.BOX('JAD HD:REAL DEAL:WVAL1!B3:H8)
=ACTIVATE("WVAL1")
=SELECT('JAD HD:REAL DEAL:WVAL1!H$5)
=COPY()
=ACTIVATE("Summary")
=SELECT('JAD HD:REAL DEAL:Summary!B$5)
print to summary sheet =PASTE.SPECIAL(3,1)
=MESSAGE(FALSE)
=ACTIVATE("Typetwo")
form of wait calculation =DIALOG.BOX('JAD HD:REAL DEAL:wait!B3:H9)
=MESSAGE(TRUE,"Recording the waiting time inputs to the
summary statistics file.")
print to summary sheet =ACTIVATE("wait")
=SELECT(!H$6)
=COPY()
=ACTIVATE("Summary")
=SELECT(!F2)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
=IF('JAD HD:REAL DEAL:wait!H7=4)
user specified wait time =INPUT("Enter the waiting time in minutes.",2,"WAIT TIME")
negative number loop =IF(B151<0)
=ALERT("The input must be a positive number.",2)
=GOTO(Typetwo!B151)
=END.IF()
print to summary sheet =SELECT(B151)
=COPY()
=ACTIVATE("Summary")
=SELECT(!B$10)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
=END.IF()
=MESSAGE(FALSE)

```

<i>use segments or not</i>	=INPUT("Do you wish to break the analysis into a series of segments along the route which will account for varying conditions (yes or no)?",2,"SEGMENTS","yes")
<i>if Poisson is used</i>	=INPUT("Do you wish to calculate the dwell time with Poisson probability theory (yes) or input a dwell time (no)?",2,"DWELL TIME","yes")
<i>no seg and Poisson calcs</i>	=IF(AND(B164="no",B165="yes"))
<i>Poisson information</i>	=DIALOG.BOX('JAD HD:REAL DEAL:pax'!B\$3:H\$17)
<i>negative number loop</i>	=IF(OR('JAD HD:REAL DEAL:pax'!H\$5<0,'JAD HD:REAL DEAL:pax'!H\$7<0,'JAD HD:REAL DEAL:pax'!H\$9<0,'JAD HD:REAL DEAL:pax'!H\$11<0,'JAD HD:REAL DEAL:pax'!H\$13,'JAD HD:REAL DEAL:pax'!H\$15<0))
	=ALERT("The input must be a positive number.",2)
	=GOTO(Typetwo!B167)
	=END.IF()
	=MESSAGE(TRUE,"Recording the Poisson inputs to the summary statistics file.")
<i>print to summary sheet</i>	=ACTIVATE("pax")
<i>pax boarding demand</i>	=SELECT(!H\$5)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(!C\$11)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("pax")
<i>pax deboarding demand</i>	=SELECT(!H\$7)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(!E\$11)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("pax")
<i>pax headway</i>	=SELECT(!H\$9)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(!G\$11)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("pax")
<i>board start up time</i>	=SELECT(!H\$11)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(!C\$12)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("pax")
<i>deboard start up time</i>	=SELECT(!H\$13)
	=COPY()
	=ACTIVATE("Summary")
	=SELECT(!E\$12)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("pax")

<i>Poisson probability</i>	=SELECT(!H\$15) =COPY() =ACTIVATE("Summary") =SELECT(!G\$12) =PASTE.SPECIAL(3,1) =ACTIVATE("Typetwo") =END.IF() =MESSAGE(FALSE)
<i>no seg and given dwell time</i>	=IF(AND(B164="no",B165="no"))
<i>given dwell time</i>	=INPUT("Enter the dwell time in seconds.",2,"DWELL TIME",45)
<i>negative number loop</i>	=IF(B213<0) =ALERT("The input must be a positive number.",2) =GOTO(Typetwo!B213) =END.IF()
<i>print to summary sheet</i>	=SELECT(B213) =COPY() =ACTIVATE("Summary") =SELECT(!D\$10) =PASTE.SPECIAL(3,1) =ACTIVATE("Typetwo") =END.IF()
<i>no segmentation</i>	=IF(B164="no")
<i>initialize counter</i>	=SET.NAME("solo",1)
<i>value of counter</i>	=solo
<i>loop to print incr if no seg</i>	=FOR("solo",1,20,1) =SELECT(Typetwo!B9) =COPY() =SELECT(OFFSET(Typetwo!G234,solo,0)) =PASTE.SPECIAL(3,1) =NEXT() =GOTO(B408) =END.IF()
<i>using segmentation</i>	=IF(B164="yes")
<i>number of segments</i>	=INPUT("Enter the number of segments.",1,"NUMBER OF SEGMENTS",3) =B237
<i>negative number loop</i>	=IF(B238<0) =ALERT("The input must be a positive number.",2) =GOTO(Typetwo!B237) =END.IF() =ACTIVATE("Typetwo") =COPY(!B230) =ACTIVATE("Summary") =SELECT(!D2) =PASTE.SPECIAL(3,1) =ACTIVATE("Typetwo") =END.IF()

```

seg and Poisson calcs      =IF(AND(B164="yes",B165="yes"))
message to explain cons&var =ALERT("Some of the required Poisson inputs are constant for
                             each segment and the rest vary across the segments.",2)
                             =DIALOG.BOX('JAD HD:REAL DEAL:pax'!B$18:H$28)
negative number loop      =IF(OR('JAD HD:REAL DEAL:pax'!H$20<0,'JAD HD:REAL
                             DEAL:pax'!H$22<0,'JAD HD:REAL DEAL:pax'!H$24<0,'JAD
                             HD:REAL DEAL:pax'!H$26<0))
                             =ALERT("The input must be a positive number.",2)
                             =GOTO(Typetwo!B252)
                             =END.IF()
                             =MESSAGE(TRUE,"Recording the Poisson inputs to the
                             summary statistics file.")
                             =ACTIVATE("pax")
pax headway               =SELECT(!H$20)
                             =COPY()
                             =ACTIVATE("Summary")
                             =SELECT(!G$11)
                             =PASTE.SPECIAL(3,1)
                             =ACTIVATE("pax")
board start up time      =SELECT(!H$22)
                             =COPY()
                             =ACTIVATE("Summary")
                             =SELECT(!C$12)
                             =PASTE.SPECIAL(3,1)
                             =ACTIVATE("pax")
deboard start up time    =SELECT(!H$24)
                             =COPY()
                             =ACTIVATE("Summary")
                             =SELECT(!E$12)
                             =PASTE.SPECIAL(3,1)
                             =ACTIVATE("pax")
Poisson probability      =SELECT(!H$26)
                             =COPY()
                             =ACTIVATE("Summary")
                             =SELECT(!G$12)
                             =PASTE.SPECIAL(3,1)
                             =ACTIVATE("Typetwo")
                             =MESSAGE(FALSE)
                             =ALERT("A dialog box will appear for each segment. Enter the
                             corresponding segment variables in each successive dialog
                             box.",2)
                             =SET.NAME("length",1)
                             =length
                             =FOR("length",1,B238,1)
                             =DIALOG.BOX('JAD HD:REAL DEAL:variables'!B$3:H$15)

```

negative number loop

```
=IF(OR('JAD HD:REAL DEAL:variables'!H$5<0,'JAD HD:REAL  
DEAL:variables'!H$7<0,'JAD HD:REAL  
DEAL:variables'!H$9<0,'JAD HD:REAL  
DEAL:variables'!H$11<0,'JAD HD:REAL  
DEAL:variables'!H$13<0))  
=ALERT("The input must be a positive number.",2)  
=GOTO(Typetwo!B288)  
=END.IF()  
=MESSAGE(TRUE,"Recording the segment variables to the  
summary statistics file.")  
='JAD HD:REAL DEAL:variables'!H$5  
=COPY(B294)  
=SELECT(OFFSET(Typetwo!C234,length,0))  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Summary")  
=SELECT(OFFSET('JAD HD:REAL  
DEAL:Summary'!$C$12,length,0))  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Typetwo")  
='JAD HD:REAL DEAL:variables'!H$7  
=COPY(B302)  
=SELECT(OFFSET(Typetwo!D234,length,0))  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Summary")  
=SELECT(OFFSET('JAD HD:REAL  
DEAL:Summary'!$E$12,length,0))  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Typetwo")  
='JAD HD:REAL DEAL:variables'!H$9  
=COPY(B310)  
=SELECT(OFFSET(Typetwo!E234,length,0))  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Summary")  
=SELECT(OFFSET('JAD HD:REAL  
DEAL:Summary'!$G$12,length,0))  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Typetwo")  
='JAD HD:REAL DEAL:variables'!H$11  
=COPY(B318)  
=SELECT(OFFSET(Typetwo!F234,length,0))  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Summary")  
=SELECT(OFFSET('JAD HD:REAL  
DEAL:Summary'!$I$12,length,0))  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("Typetwo")  
='JAD HD:REAL DEAL:variables'!H$13  
=COPY(B326)  
=SELECT(OFFSET(Typetwo!G234,length,0))
```

```

= PASTE.SPECIAL(3,1)
= ACTIVATE("Summary")
= SELECT(OFFSET('JAD HD:REAL
DEAL:Summary'!$K$12,length,0))
= PASTE.SPECIAL(3,1)
= ACTIVATE("Typetwo")
= MESSAGE(FALSE)
= NEXT()
= SUM(C235:C311)
segs too short = IF(B336<'JAD HD:REAL DEAL:LHM2'!H$19)
= ALERT("The sum of the segment lengths is less than the route
length.",3)
= DIALOG.BOX('JAD HD:REAL DEAL:LHM2'!B$17:H$20)
= GOTO(B287)
= END.IF()
segs too long = IF(B336>'JAD HD:REAL DEAL:LHM2'!H$19)
= ALERT("The sum of the segment lengths is greater than the
route length.",3)
= DIALOG.BOX('JAD HD:REAL DEAL:LHM2'!B$17:H$20)
= GOTO(B287)
= END.IF()
end seg and Poisson loop = END.IF()
seg and given dwell time = IF(AND(B164="yes",B165="no"))
= ALERT("A dialog box will appear for each segment. Enter the
corresponding segment variables in each successive dialog
box.",2)
= SET.NAME("length",1)
=length
= FOR("length",1,B238,1)
= DIALOG.BOX('JAD HD:REAL DEAL:variables'!B$16:H$24)
negative number loop = IF(OR('JAD HD:REAL DEAL:variables'!H$18<0,'JAD HD:REAL
DEAL:variables'!H$20<0,'JAD HD:REAL
DEAL:variables'!H$22<0))
= ALERT("The input must be a positive number.",2)
= GOTO(Typetwo!B353)
= END.IF()
= MESSAGE(TRUE,"Recording the segment variables to the
summary statistics file.")
='JAD HD:REAL DEAL:variables'!H$18
= COPY(B359)
= SELECT(OFFSET(Typetwo!C234,length,0))
= PASTE.SPECIAL(3,1)
= ACTIVATE("Summary")
= SELECT(OFFSET(!C$12,length,0))
= PASTE.SPECIAL(3,1)
= ACTIVATE("Typetwo")
='JAD HD:REAL DEAL:variables'!H$20
= COPY(B367)
= SELECT(OFFSET(Typetwo!F234,length,0))

```

```

=PASTE.SPECIAL(3,1)
=ACTIVATE("Summary")
=SELECT(OFFSET(!I$12,length,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
='JAD HD:REAL DEAL:variables!H$22
=COPY(B375)
=SELECT(OFFSET(Typetwo!G234,length,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Summary")
=SELECT(OFFSET(!K$12,length,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
=NEXT()
=SUM(C235:C311)
segs too short
=IF(B384<'JAD HD:REAL DEAL:LHM2!H$19)
=ALERT("The sum of the segment lengths is less than the route
length.",3)
=DIALOG.BOX('JAD HD:REAL DEAL:LHM2!B$17:H$20)
=GOTO(B352)
=END.IF()
segs too long
=IF(B384>'JAD HD:REAL DEAL:LHM2!H$19)
=ALERT("The sum of the segment lengths is greater than the
route length.",3)
=DIALOG.BOX('JAD HD:REAL DEAL:LHM2!B$17:H$20)
=GOTO(B352)
=END.IF()
given dwell time
negative number loop
=INPUT("Enter the dwell time in seconds.",2,"DWELL TIME",45)
=IF(B395<0)
=ALERT("The input must be a positive number.",2)
=GOTO(Typetwo!B395)
=END.IF()
print to summary sheet
=SELECT(B395)
=COPY()
=ACTIVATE("Summary")
=SELECT(!D10)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
=END.IF()
=MESSAGE(FALSE)
wait time loop
half headway (wait)
=IF('JAD HD:REAL DEAL:wait!H$7=1)
='JAD HD:REAL DEAL:LHM2!H8/2
=END.IF()
=IF('JAD HD:REAL DEAL:wait!H$7=2)
='JAD HD:REAL DEAL:LHM2!H8^0.5
=END.IF()
sq root of headway (wait)
=IF('JAD HD:REAL DEAL:wait!H$7=3)

```



```

=IF(OR('JAD HD:REAL DEAL:LHM2'!H8>30,'JAD HD:REAL
DEAL:LHM2'!H8=30))
='JAD HD:REAL DEAL:LHM2'!H8/2
=ELSE()
='JAD HD:REAL DEAL:LHM2'!H8^0.5
=END.IF()
=END.IF()
switch to rail macro =IF(OR('JAD HD:REAL DEAL:LHM2'!H6=3,'JAD HD:REAL
DEAL:LHM2'!H6=4))
=OPEN("railtwo",TRUE)
=MESSAGE(TRUE,"Opening the necessary files.")
=RUN('JAD HD:REAL DEAL:railtwo'!B2)
goto end when return from rail =GOTO(B2999)
close switch to rail loop =END.IF()
if linehaul mode local bus =IF('JAD HD:REAL DEAL:LHM2'!H6=1)
=IF('JAD HD:REAL DEAL:WVAL1'!H6=1)
turn off screen updating =ECHO(FALSE)
=MESSAGE(TRUE,"Opening the necessary files.")
=ACTIVATE("Opener2")
open files for willing w value =RUN(opener2!B12)
return control to this macro =ACTIVATE("Typetwo")
=MESSAGE(FALSE)
=ECHO(FALSE)
access mode inputs if given =DIALOG.BOX('JAD HD:REAL DEAL:WACC1'!B3:H12)
w value =IF('JAD HD:REAL DEAL:WACC1'!$H$6=2)
=SET.NAME("walk",1)
=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:WACC1'!$I$8,'JAD
HD:REAL DEAL:WACC1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:WACC1'!$B$3:$H$12)
=NEXT()
=END.IF()
=IF('JAD HD:REAL DEAL:WACC1'!$H$6=3)
=SET.NAME("walk",1)
=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:WACC1'!$I$9,'JAD
HD:REAL DEAL:WACC1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:WACC1'!$B$3:$H$12)
=NEXT()
=END.IF()
=IF('JAD HD:REAL DEAL:WACC1'!$H$6=4)
=SET.NAME("walk",1)
=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:WACC1'!$I$10,'JAD
HD:REAL DEAL:WACC1'!$H$8)
=SET.NAME("walk",walk+1)

```

```

=DIALOG.BOX('JAD HD:REAL DEAL:WACC1'!$B$3:$H$12)
=NEXT()
=END.IF()
negative number loop =IF(OR('JAD HD:REAL DEAL:WACC1'!H$8<0,'JAD HD:REAL
DEAL:WACC1'!H284<0))
=ALERT("The input must be a positive number.",2)
=GOTO(Typetwo!B436)
=END.IF()
print to summary sheet =MESSAGE(TRUE,"Recording the walk access inputs to the
mode summary statistics file.")
=ACTIVATE("WACC1")
=SELECT('JAD HD:REAL DEAL:WACC1'!H$5)
=COPY()
=ACTIVATE("Summary")
=SELECT(!B$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("WACC1")
rate =SELECT(!H$8)
=COPY()
=ACTIVATE("Summary")
=SELECT(!D$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("WACC1")
w value =SELECT(!H$10)
=COPY()
=ACTIVATE("Summary")
=SELECT(!F$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
=ELSE()
turn off screen updating =ECHO(FALSE)
=MESSAGE(TRUE,"Opening the necessary files.")
=ACTIVATE("Opener2")
open files for actual distance =RUN(opener2!B19)
=MESSAGE(FALSE)
return control to this macro =ACTIVATE("Typetwo")
inputs if given actual distance =DIALOG.BOX('JAD HD:REAL DEAL:ACCDIST1'!$B$3:$H$12)
=IF('JAD HD:REAL DEAL:ACCDIST1'!$H$6=2)
=SET.NAME("walk",1)
=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:ACCDIST1'!$I$8,'JAD
HD:REAL DEAL:ACCDIST1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:ACCDIST1'!$B$3:$H$12)
=NEXT()
=END.IF()
=IF('JAD HD:REAL DEAL:ACCDIST1'!$H$6=3)
=SET.NAME("walk",1)

```

```

=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:ACCDIST1'!$I$9,'JAD
HD:REAL DEAL:ACCDIST1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:ACCDIST1'!$B$3:$H$12)
=NEXT()
=END.IF()
=IF('JAD HD:REAL DEAL:ACCDIST1'!$H$6=4)
=SET.NAME("walk",1)
=WHILE(walk=1)
=FORMULA('JAD HD:REAL DEAL:ACCDIST1'!$I$10,'JAD
HD:REAL DEAL:ACCDIST1'!$H$8)
=SET.NAME("walk",walk+1)
=DIALOG.BOX('JAD HD:REAL DEAL:ACCDIST1'!$B$3:$H$12)
=NEXT()
=END.IF()
negative number loop =IF(OR('JAD HD:REAL DEAL:ACCDIST1'!H$8<0,'JAD
HD:REAL DEAL:ACCDIST1'!H$10<0))
=ALERT("The input must be a positive number.",2)
=GOTO(Typetwo!B492)
=END.IF()
print to summary sheet =MESSAGE(TRUE,"Recording the walk access inputs to the
mode summary statistics file.")
=ACTIVATE("ACCDIST1")
=SELECT('JAD HD:REAL DEAL:ACCDIST1'!H$5)
=COPY()
=ACTIVATE("Summary")
=SELECT(!B$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("ACCDIST1")
rate =SELECT(!H$8)
=COPY()
=ACTIVATE("Summary")
=SELECT(!D$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("ACCDIST1")
actual distance =SELECT(!H$10)
=COPY()
=ACTIVATE("Summary")
=SELECT(!H$6)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
=END.IF()
end walk input loop =IF('JAD HD:REAL DEAL:WVAL1'!H6=1)
egress inputs if willing... =DIALOG.BOX('JAD HD:REAL DEAL:Egress1'!B3:H8)
egress mode choices =ACTIVATE("EGRESS1")
print to summary sheet =SELECT(!H$5)
=COPY()

```

```

=ACTIVATE("Summary")
=SELECT(!B$20)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
begin walk loop
egress mode inputs if walk
negative number loop
=IF('JAD HD:REAL DEAL:Egress1'!H6=1)
=DIALOG.BOX('JAD HD:REAL DEAL:WALKEGR'!B3:H9)
=IF(OR('JAD HD:REAL DEAL:WALKEGR'!H$5<0,'JAD
HD:REAL DEAL:WALKEGR'!H$7<0))
=ALERT("The inputs must be positive numbers.",2)
=GOTO(Typetwo!B552)
=END.IF()
=MESSAGE(TRUE,"Recording the walk egress inputs to the
summary statistics file.")
print to summary sheet
=ACTIVATE("walkegr")
=SELECT(!H$5)
=COPY()
=ACTIVATE("Summary")
walk egress rate
=SELECT(!D$20)
=PASTE.SPECIAL(3,1)
=ACTIVATE("walkegr")
=SELECT(!H$7)
=COPY()
=ACTIVATE("Summary")
w value
=SELECT(!F$20)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
end walk loop
=END.IF()
=MESSAGE(FALSE)
begin bus/walk loop
egress mode inputs if
bus/walk
negative number loop
=IF('JAD HD:REAL DEAL:Egress1'!H6=2)
=DIALOG.BOX('JAD HD:REAL DEAL:WBUSDIST1'!B3:H15)
=IF(OR('JAD HD:REAL DEAL:WBUSDIST1'!H$5<0,'JAD
HD:REAL DEAL:WBUSDIST1'!H$7<0,'JAD HD:REAL
DEAL:WBUSDIST1'!H$9<0))
=ALERT("The inputs must be positive numbers.",2)
=GOTO(Typetwo!B574)
=END.IF()
=MESSAGE(TRUE,"Recording the bus egress inputs to the
summary statistics file.")
print to summary sheet
=ACTIVATE("WBUSDIST1")
=SELECT(!H$5)
=COPY()
=ACTIVATE("Summary")
egress bus rate
=SELECT(!D$21)
=PASTE.SPECIAL(3,1)
=ACTIVATE("WBUSDIST1")
=SELECT(!H$7)
=COPY()

```

```

egress bus w value      =ACTIVATE("Summary")
                        =SELECT(!F$21)
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("WBUSDIST1")
                        =SELECT(!H$9)
                        =COPY()
egress bus headway     =ACTIVATE("Summary")
                        =SELECT(!J$21)
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("WBUSDIST1")
walk rate from egress bus =SELECT(!H$11)
                        =COPY()
                        =ACTIVATE("Summary")
                        =SELECT(!N$21)
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("WBUSDIST1")
walk w value from egress bus =SELECT(!H$13)
                        =COPY()
                        =ACTIVATE("Summary")
                        =SELECT(!P$21)
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typetwo")
end bus/walk loop      =END.IF()
                        =MESSAGE(FALSE)
begin combo input loop =IF('JAD HD:REAL DEAL:Egress1'!H6=3)
walk                   =DIALOG.BOX('JAD HD:REAL DEAL:WWALKEGR2'!B12:H20)
negative number loop   =IF(OR('JAD HD:REAL DEAL:WWALKEGR2'!H$14<0,'JAD
                        HD:REAL DEAL:WWALKEGR2'!H$16<0,'JAD HD:REAL
                        DEAL:WWALKEGR2'!H$18<0))
                        =ALERT("The inputs must be positive numbers.",2)
                        =GOTO(Typetwo!B614)
                        =END.IF()
bus/walk               =DIALOG.BOX('JAD HD:REAL DEAL:WBUSEGR2'!B3:H17)
negative number loop   =IF(OR('JAD HD:REAL DEAL:WBUSEGR2'!H$5<0,'JAD
                        HD:REAL DEAL:WBUSEGR2'!H$7<0,'JAD HD:REAL
                        DEAL:WBUSEGR2'!H$9<0,'JAD HD:REAL
                        DEAL:WBUSEGR2'!H$11<0))
                        =ALERT("The inputs must be positive numbers.",2)
                        =GOTO(Typetwo!B619)
                        =END.IF()
                        =IF(OR('JAD HD:REAL DEAL:WWALKEGR2'!H14+'JAD
                        HD:REAL DEAL:WBUSEGR2'!H5>100,'JAD HD:REAL
                        DEAL:WWALKEGR2'!H14+'JAD HD:REAL
                        DEAL:WBUSEGR2'!H5<100))
check that % add to 100 =ALERT("Egress mode percentages must add to 100%.",2)
                        =GOTO(B614)
                        =END.IF()
                        =MESSAGE(TRUE,"Recording the combination egress inputs
                        to the summary statistics file.")

```

	=ACTIVATE("WWALKEGR2")
	=SELECT(!H\$14)
	=COPY()
<i>walk egress percent</i>	=ACTIVATE("Summary")
	=SELECT(!L\$20)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("WWALKEGR2")
	=SELECT(!H\$16)
	=COPY()
	=ACTIVATE("Summary")
<i>walk egress rate</i>	=SELECT(!D\$20)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("WWALKEGR2")
	=SELECT(!H\$18)
	=COPY()
	=ACTIVATE("Summary")
<i>walk egress w value</i>	=SELECT(!F\$20)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("WBUSEGR2")
	=SELECT(!H\$5)
	=COPY()
	=ACTIVATE("Summary")
<i>bus egress percent</i>	=SELECT(!L\$21)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("WBUSEGR2")
	=SELECT(!H\$7)
	=COPY()
	=ACTIVATE("Summary")
<i>bus egress rate</i>	=SELECT(!D\$21)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("WBUSEGR2")
	=SELECT(!H\$9)
	=COPY()
	=ACTIVATE("Summary")
<i>bus egress w value</i>	=SELECT(!F\$21)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("WBUSEGR2")
	=SELECT(!H\$11)
	=COPY()
	=ACTIVATE("Summary")
<i>bus egress headway</i>	=SELECT(!J\$21)
	=PASTE.SPECIAL(3,1)
	=ACTIVATE("WBUSEGR2")
	=SELECT(!H\$13)
	=COPY()
	=ACTIVATE("Summary")
<i>walk rate from egress bus</i>	=SELECT(!N\$21)

```

= PASTE.SPECIAL(3,1)
= ACTIVATE("WBUSEGR2")
= SELECT(!H$15)
= COPY()
= ACTIVATE("Summary")
walk w value from egress bus = SELECT(!P$21)
= PASTE.SPECIAL(3,1)
= ACTIVATE("Typetwo")
end combo input loop = END.IF()
= MESSAGE(FALSE)
switch to actual egress dist = ELSE()
egress mode choices = DIALOG.BOX('JAD HD:REAL DEAL:Egress1'!B3:H8)
print to summary sheet = ACTIVATE("EGRESS1")
= SELECT(!H$5)
= COPY()
= ACTIVATE("Summary")
= SELECT(!B$20)
= PASTE.SPECIAL(3,1)
= ACTIVATE("Typetwo")
begin walk loop = IF('JAD HD:REAL DEAL:Egress1'!H6=1)
walk distance inputs = DIALOG.BOX('JAD HD:REAL DEAL:WALKEGR'!B10:H16)
negative number loop = IF(OR('JAD HD:REAL DEAL:WALKEGR'!H$12<0,'JAD
HD:REAL DEAL:WALKEGR'!H$14<0))
= ALERT("The inputs must be positive numbers.",2)
= GOTO(Typetwo!B696)
= END.IF()
= MESSAGE(TRUE,"Recording the walk egress inputs to the
summary statistics file.")
print to summary sheet = ACTIVATE("walkegr")
= SELECT(!H$12)
= COPY()
= ACTIVATE("Summary")
walk egress rate = SELECT(!D$20)
= PASTE.SPECIAL(3,1)
= ACTIVATE("walkegr")
= SELECT(!H$14)
= COPY()
= ACTIVATE("Summary")
walk distance = SELECT(!H$20)
= PASTE.SPECIAL(3,1)
= ACTIVATE("Typetwo")
end walk loop = END.IF()
= MESSAGE(FALSE)
begin bus/walk loop = IF('JAD HD:REAL DEAL:Egress1'!H6=2)
bus/walk distance inputs = DIALOG.BOX('JAD HD:REAL DEAL:BusEgr1'!B3:H15)
negative number loop = IF(OR('JAD HD:REAL DEAL:BusEgr1'!H$5<0,'JAD HD:REAL
DEAL:BusEgr1'!H$7<0,'JAD HD:REAL
DEAL:BusEgr1'!H$9<0))

```

```

=ALERT("The inputs must be positive numbers.",2)
=GOTO(Typetwo!B718)
=END.IF()
=MESSAGE(TRUE,"Recording the bus egress inputs to the
summary statistics file.")
print to summary sheet    =ACTIVATE("BUSEGR1")
                          =SELECT(!H$5)
                          =COPY()
                          =ACTIVATE("Summary")
egress bus rate          =SELECT(!D$21)
                          =PASTE.SPECIAL(3,1)
                          =ACTIVATE("BUSEGR1")
                          =SELECT(!H$7)
                          =COPY()
                          =ACTIVATE("Summary")
egress bus distance      =SELECT(!H$21)
                          =PASTE.SPECIAL(3,1)
                          =ACTIVATE("BUSEGR1")
                          =SELECT(!H$9)
                          =COPY()
                          =ACTIVATE("Summary")
egress bus headway       =SELECT(!J$21)
                          =PASTE.SPECIAL(3,1)
                          =ACTIVATE("BUSEGR1")
                          =SELECT(!H$11)
                          =COPY()
                          =ACTIVATE("Summary")
walk rate from egress bus =SELECT(!N$21)
                          =PASTE.SPECIAL(3,1)
                          =ACTIVATE("BUSEGR1")
                          =SELECT(!H$13)
                          =COPY()
                          =ACTIVATE("Summary")
walk dist from egress bus =SELECT(!R$21)
                          =PASTE.SPECIAL(3,1)
                          =ACTIVATE("Typetwo")
end bus loop             =END.IF()
                          =MESSAGE(FALSE)
begin combo input loop   =IF('JAD HD:REAL DEAL:Egress1'!H6=3)
walk                     =DIALOG.BOX('JAD HD:REAL DEAL:WalkEgr2'!B12:H20)
negative number loop     =IF(OR('JAD HD:REAL DEAL:WalkEgr2'!H$14<0,'JAD
                          HD:REAL DEAL:WalkEgr2'!H$16<0,'JAD HD:REAL
                          DEAL:WalkEgr2'!H$18<0))
                          =ALERT("The inputs must be positive numbers.",2)
                          =GOTO(Typetwo!B758)
                          =END.IF()
bus/walk                  =DIALOG.BOX('JAD HD:REAL DEAL:BusEgr2'!B3:H17)

```


negative number loop

```
=IF(OR('JAD HD:REAL DEAL:BusEgr2'!H$5<0,'JAD HD:REAL  
DEAL:BusEgr2'!H$7<0,'JAD HD:REAL  
DEAL:BusEgr2'!H$9<0,'JAD HD:REAL  
DEAL:BusEgr2'!H$11<0))  
=ALERT("The inputs must be positive numbers.",2)  
=GOTO(Typetwo!B763)  
=END.IF()
```

check that % add to 100

```
=IF(OR('JAD HD:REAL DEAL:WalkEgr2'!H14+'JAD HD:REAL  
DEAL:BusEgr2'!H5>100,'JAD HD:REAL  
DEAL:WalkEgr2'!H14+'JAD HD:REAL  
DEAL:BusEgr2'!H5<100))  
=ALERT("Egress mode percentages must add to 100%.",2)  
=GOTO(B758)  
=END.IF()
```

walk egress percent

```
=MESSAGE(TRUE,"Recording the combination egress inputs  
to the summary statistics file.")  
=ACTIVATE("WALKEGR2")  
=SELECT(!H$14)  
=COPY()  
=ACTIVATE("Summary")
```

walk egress rate

```
=SELECT(!L$20)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("WALKEGR2")  
=SELECT(!H$16)  
=COPY()  
=ACTIVATE("Summary")
```

walk egress distance

```
=SELECT(ID$20)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("WALKEGR2")  
=SELECT(!H$18)  
=COPY()  
=ACTIVATE("Summary")
```

bus egress percent

```
=SELECT(!H$20)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("BusEgr2")  
=SELECT(!H$5)  
=COPY()  
=ACTIVATE("Summary")
```

bus egress rate

```
=SELECT(!L$21)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("BusEgr2")  
=SELECT(!H$7)  
=COPY()  
=ACTIVATE("Summary")
```

```
=SELECT(ID$21)  
=PASTE.SPECIAL(3,1)  
=ACTIVATE("BusEgr2")  
=SELECT(!H$9)
```

```

=COPY()
=ACTIVATE("Summary")
bus egress distance =SELECT(!F$21)
=PASTE.SPECIAL(3,1)
=ACTIVATE("BusEgr2")
=SELECT(!H$11)
=COPY()
=ACTIVATE("Summary")
bus egress headway =SELECT(!J$21)
=PASTE.SPECIAL(3,1)
=ACTIVATE("BusEgr2")
=SELECT(!H$13)
=COPY()
=ACTIVATE("Summary")
walk rate from egress bus =SELECT(!N$21)
=PASTE.SPECIAL(3,1)
=ACTIVATE("BusEgr2")
=SELECT(!H$15)
=COPY()
=ACTIVATE("Summary")
walk dist from egress bus =SELECT(!R$21)
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
=END.IF()
end combo input loop =END.IF()
end loop for egress inputs =MESSAGE(FALSE)
=ECHO(FALSE)
=SET.NAME("answer",0)
=SET.NAME("seg",1)
skip counter stuff for solo =IF(B$164="no")
=GOTO(B846)
=END.IF()
=WHILE(OR(seg<B238,seg=B238))
=MESSAGE(FALSE)
=answer
=seg
=COPY(B840)
=ACTIVATE("WORKSHEETS")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEETS'!$A$3,0,answer))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
=SET.NAME("count",0)
=OFFSET(Typetwo!G234,seg,0)
initialize counter to zero =WHILE(OR(B20+count*B847<B31,B20+count*B84
7=B31))
=copy(B840)
=ACTIVATE("WORKSHEETS")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEETS'!$A$3,0,answer))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
=SET.NAME("count",0)
=OFFSET(Typetwo!G234,seg,0)
test conditions to continue =WHILE(OR(B20+count*B847<B31,B20+count*B84
7=B31))
loop =copy(B840)
display spacing =ACTIVATE("WORKSHEETS")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEETS'!$A$3,0,answer))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
=SET.NAME("count",0)
=OFFSET(Typetwo!G234,seg,0)

```

```

=MESSAGE(TRUE,"Computing the access and egress costs.")
if stmt if willing to travel =IF('JAD HD:REAL DEAL:WVAL1'!H6=1)
walk distance =(((JAD HD:REAL
DEAL:WACC1'!H10^2)+(Typetwo!B849/2)^2)^0.5)
walk time min =B852/('JAD HD:REAL DEAL:WACC1'!H8*60)
begin if loop =IF('JAD HD:REAL DEAL:Egress1'!H6=1)
walk egress distance w value =(((JAD HD:REAL
DEAL:WALKEGR'!H7^2)+(Typetwo!B849/2)^2)^0.5)
walk egress time min =B855/('JAD HD:REAL DEAL:WALKEGR'!H5*60)
acc/egr time if walk =B853+B856
end walk w value calculations =END.IF()
begin bus/walk w value calcs =IF('JAD HD:REAL DEAL:Egress1'!H6=2)
bus/walk w value calcs =(('JAD HD:REAL DEAL:WBUSDIST1'!H7)/('JAD HD:REAL
DEAL:WBUSDIST1'!H5/60))+(((JAD HD:REAL
DEAL:WBUSDIST1'!H13^2)+(Typetwo!B849/2)^2)^0.5)/('JAD
HD:REAL DEAL:WBUSDIST1'!H11*60))
=JAD HD:REAL DEAL:WBUSDIST1'!H9^0.5
wait time for transfer =B860+B861
bus walk,wait,ride time =B853+B860+B861
acc/egr time if bus/walk =END.IF()
end bus/walk w value loop =IF('JAD HD:REAL DEAL:Egress1'!H6=3)
begin combo w value loop =(((JAD HD:REAL
DEAL:WWALKEGR2'!H18^2)+(Typetwo!B849/2)^2)^0.5)
walk w value =B866/('JAD HD:REAL DEAL:WWALKEGR2'!H16*60)
acc/egr w value walk time =B853+B867
bus/walk egress w value time =(('JAD HD:REAL DEAL:WBUSEGR2'!H9)/('JAD HD:REAL
DEAL:WBUSEGR2'!H7/60))+(((JAD HD:REAL
DEAL:WBUSEGR2'!H15^2)+(Typetwo!B849/2)^2)^0.5)/('JAD
HD:REAL DEAL:WBUSEGR2'!H13*60))
=JAD HD:REAL DEAL:WBUSEGR2'!H11^0.5
wait time for transfer =B869+B870
bus wait,ride,walk egress =B853+B869+B870
acc/egr w time bus/walk =(('JAD HD:REAL
DEAL:WWALKEGR2'!H14/100)*Typetwo!B867)+('JAD
HD:REAL DEAL:WBUSEGR2'!H5/100)*Typetwo!B871)
weighted w value egr time =(('JAD HD:REAL
DEAL:WWALKEGR2'!H14/100)*Typetwo!B868)+('JAD
HD:REAL DEAL:WBUSEGR2'!H5/100)*Typetwo!B872)
end combo loop =END.IF()
if providing actual distance =ELSE()
actual access distance time (min) =('JAD HD:REAL DEAL:ACCDIST1'!H10)/('JAD HD:REAL
DEAL:ACCDIST1'!H8*60))
begin walk distance loop =IF('JAD HD:REAL DEAL:Egress1'!H6=1)
walk egress time =JAD HD:REAL DEAL:WALKEGR'!H14/('JAD HD:REAL
DEAL:WALKEGR'!H12*60)
acc/egr time if walk =B877+B879
end walk distance egress loop =END.IF()
begin bus/walk distance loop =IF('JAD HD:REAL DEAL:Egress1'!H6=2)
bus/walk distance time =('JAD HD:REAL DEAL:BusEgr1'!H7)/('JAD HD:REAL
DEAL:BusEgr1'!H5/60))+('JAD HD:REAL
DEAL:BusEgr1'!H13/('JAD HD:REAL DEAL:BusEgr1'!H11*60))

```

```

wait time for transfer =JAD HD:REAL DEAL:BusEgr1!H9^0.5
wait,ride,walk bus transfer =B883+B884
acc/egr time if bus/walk =B877+B883+B884
end bus/walk distance loop =END.IF()
begin combo distance loop =IF('JAD HD:REAL DEAL:Egress1!H6=3)
walk distance time =('JAD HD:REAL DEAL:WalkEgr2!H18/('JAD HD:REAL
DEAL:WalkEgr2!H16*60))
acc/egr time if walk =B877+B889
bus/walk egress min =('JAD HD:REAL DEAL:BusEgr2!H9/('JAD HD:REAL
DEAL:BusEgr2!H7/60))+('JAD HD:REAL
DEAL:BusEgr2!H15/('JAD HD:REAL DEAL:BusEgr2!H13*60))
transfer wait time ='JAD HD:REAL DEAL:BusEgr2!H11^0.5
wait,ride,walk bus transfer =B891+B892
acc/egr time bus/walk dist =B877+B891+B892
weighted given distance =('JAD HD:REAL
egress time DEAL:WalkEgr2!H14/100)*(Typetwo!B889))+('JAD HD:REAL
DEAL:BusEgr2!H5/100)*(Typetwo!B893))
weighted given distance time =('JAD HD:REAL
for acc,wait,egr DEAL:WalkEgr2!H14/100)*(Typetwo!B890))+('JAD HD:REAL
DEAL:BusEgr2!H5/100)*(Typetwo!B894))
end combo distance loop =END.IF()
end actual dist or w value loop =END.IF()
no segmentation =IF(B164="no")
=MESSAGE(TRUE,"Recording the access costs to the results
file.")
show end product =ECHO(FALSE)
=ACTIVATE("Typetwo")
=SELECT(B849)
=COPY()
=ACTIVATE("WORKSHEET")
print the spacing on screen =SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET!A$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
walk access print loop =IF('JAD HD:REAL DEAL:WVAL1!H6=1)
willing to travel =SELECT(B853)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET!B$3,count,0))
=PASTE.SPECIAL(3,1)
=ACTIVATE("Typetwo")
switch to actual distance =ELSE()
actual walk distance =SELECT(B877)
=COPY()
=ACTIVATE("WORKSHEET")
=SELECT(OFFSET('JAD HD:REAL
DEAL:WORKSHEET!B$3,count,0))
=PASTE.SPECIAL(3,1)

```

```

end walk print loop      =ACTIVATE("Typetwo")
                        =END.IF()
                        =MESSAGE(TRUE,"Recording the waiting time costs to the
                        results file.")
print wait time          =IF('JAD HD:REAL DEAL:wait'!$H$7=1)
half headway            =SELECT(B409)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!C$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typetwo")
                        =END.IF()
sq root headway         =IF('JAD HD:REAL DEAL:wait'!$H$7=2)
print wait time         =SELECT(B412)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!C$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typetwo")
                        =END.IF()
30 min criteria         =IF('JAD HD:REAL DEAL:wait'!$H$7=3)
                        =IF(OR('JAD HD:REAL DEAL:LHM1'!H$9>30,'JAD HD:REAL
                        DEAL:LHM1'!H$9=30))
print wait time         =SELECT(B416)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!C$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typetwo")
                        =ELSE()
print wait time         =SELECT(B418)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!C$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typetwo")
                        =END.IF()
                        =END.IF()
user specified time     =IF('JAD HD:REAL DEAL:wait'!$H$7=4)
print wait time         =SELECT(B151)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!C$3,count,0))
                        =PASTE.SPECIAL(3,1)

```

```

end wait print loop      =ACTIVATE("Typetwo")
                        =END.IF()
                        =MESSAGE(TRUE,"Recording the waiting time costs to the
                        results file.")
walk egress print loop  =IF('JAD HD:REAL DEAL:Egress1'!H$6=1)
print w value if willing walk
egress                  =IF('JAD HD:REAL DEAL:WVAL1'!H$6=1)

                        =SELECT(B856)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!F$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typetwo")
                        =ELSE()
print if walk distance  =SELECT(B879)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!F$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typetwo")
end w value or distance loop =END.IF()
end walk egress loop    =END.IF()
bus/walk egress print loop =IF('JAD HD:REAL DEAL:Egress1'!H$6=2)
                        =IF('JAD HD:REAL DEAL:WVAL1'!H$6=1)
print bus/walk w value  =SELECT(B862)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!F$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typetwo")
                        =ELSE()
print bus/walk distance =SELECT(B885)
                        =COPY()
                        =ACTIVATE("WORKSHEET")
                        =SELECT(OFFSET('JAD HD:REAL
                        DEAL:WORKSHEET'!F$3,count,0))
                        =PASTE.SPECIAL(3,1)
                        =ACTIVATE("Typetwo")
end w value or distance loop =END.IF()
end bus/walk loop       =END.IF()
combo egress print loop =IF('JAD HD:REAL DEAL:Egress1'!H$6=3)
                        =IF('JAD HD:REAL DEAL:WVAL1'!H$6=1)
print combo w value     =SELECT(B873)
                        =COPY()
                        =ACTIVATE("WORKSHEET")

```