Public Transportation Modeling

New Mexico High School
Supercomputing Challenge
Final Report
April 2, 2003

Team #025
Bosque School

Team Members
Ryan J. Merrick
Nathan L. Perkins

Teachers
Debra M. Loftin

Project Advisor
Ian Ford

Table of Contents

Executive Summary ... 1

Introduction ... 2

- The Problem ... 2
- Statistics ... 2
- The Solution ... 3
- The Program ... 4

Project Description ... 5

The Code ... 6

Results ... 7

Conclusions ... 9

Recommendations ... 10

Acknowledgements ... 12

References ... 13

Appendix A: Classes ... 14

Appendix B: Program Walkthrough (Flow chart) ... 16

Appendix C: Example World File ... 17

Appendix D: Graphs ... 20

Executive Summary

The concept of efficient, environmentally clean and safe public transportation systems becomes more appealing each day, as roads grow more congested with urban traffic resulting from suburban sprawl. In an age where technology has created newer and better public transportation systems, governments are now reconsidering their benefits in various urban areas.

This project attempts to tackle the complex issues of modeling public transportation systems in different urban, suburban and intercity scenarios. The program deals primarily with track based systems, loading different scenarios and layouts from the primary world file. The program than runs a time based simulation, moving trains and people from stop to stop. Throughout the program, satisfaction and frustration statistics are recorded, as well as financial estimates to test the different benefits and drawbacks of the systems.

The program was tested with three scenarios based on Albuquerque and the surrounding area. Financially, the low maintenance costs of a single track from Santa Fe to Albuquerque make it the most appealing to management, while passengers are most satisfied with a track that circles the urban centers (downtown and the university), although this scenario will not earn enough even to pay the maintenance. Overall, the program suggests that the Albuquerque area is not ready for public transportation, both because of high financial costs and lack of user satisfaction.

Upon review of the program and model results, it appears that the program does not truly capture the complex nature of transportation. To make the model manageable,

many real world values had to be ignored, causing the program to lose its relevance in the real world.

Introduction

The Problem

Urban traffic is slow and congested in large cities as gasoline-powered vehicles crawl along carrying people (often only single occupancy) to nearby destinations. This massive amount of vehicular traffic depletes our petroleum resources, pollutes our air and causes frustrations for people trying to navigate through a city. The constant need for maintenance of roads drains public fiscal resources. The automobile is a costly, high maintenance machine that is not only damaging the environment through pollution and waste of our natural resources, but is a dangerous mode of transportation. Automobile accidents kill tens of thousands of Americans a year. Also, with the ever-increasing number of automobiles on the world's streets and highways, traffic congestion is worsening, especially in large cities, and people are spending more time commuting than ever before. While the automoboile has become an essential piece of our culture as well as our most used form of transportation, the automobile has caused many problems that we feel can be solved, in many cases, by various methods of public mass transportation.

Statistics

The United States Bureau of Public Transportation Statistics has some interesting information available regarding private transportation in comparison with public transportation. Today, road maintenance costs are reaching record highs of over

$32 billion in the United States annually. Accidents are increasing as well, with almost 50,000 deaths a year. Even automobile maintenance and gasoline costs are increasing, approaching an average of $10,000 per person per vehicle in the country while the cost of public transportation is, at a base fare of $1, a mere $1,321 per person traveling a comparable distances. When one person utilizes public transportation instead of driving himself, almost 600 gallons of gas are saved per year. In general, almost all statistics seem to suggest that public transportation is better in regard to safety, cost and the environment.

The Solution

In recent years, the speed of public transportation has been increasing exponentially. Public transportation now appears to be a feasible replacement for the automobile, allowing people to navigate around a city quickly. Public transportation is more efficient because it moves more people per vehicle than automobiles. In fact, the average number of riders in any given automobiles on American roads is 1.8. Public transportation, in spite of a high initial cost and infrastructure maintenance fees, cab end up saving money for the consumer as well as for the administrator of the public transportation system due to efficiency of the system. Also, public transportation, having been used for almost one hundred years, has proven much safer than cars. For instance in New York city, in more than ninety years of operation, with millions of riders served, less than one hundred thirty people have been killed and less than three hundred have been injured in accidents on the subway lines which stretch throughout four of New York’s five boroughs. The safety aspect brings up a major reason for this project: in addition to the economic and environmental interest that people have in public transportation, of great importance is the human impact – the safety of the riders and the time that they spend traveling. Human life and time are two valuable resources that need to be measured when studying a subject that is as much a part of human life as is transportation. It was determined that modes of public transportation could be solutions to many, but not all, of the transportation problems that are facing humans today. However, a model had to be created in order to validate the solution of public transportation and test its effectiveness.

The Program

Ultimately, the purpose of this project was to model systems of public transportation in order to find their success – the degree of success being measured in terms of financial cost, environmental impact, efficiency, and customer satisfaction. The model was a complex undertaking; however, the data that it was slated to provide simulates public transportation scenarios and within these simulations simulates vehicle movement, customer satisfaction, time spent traveling, infrastructure costs, and maintenance costs. Its purpose was not to prove that public transportation is the only answer to traffic problems, nor was it to prove that public transportation is a feasible solution everywhere. The model was created in order to simulate modes of public transportation and to determine how well they perform. The rubric, introduced above and explained later in detail, describes what constitutes the performance of a public transportation system.

Project Description

With the growing issue of transportation problems (congestion, accidents and pollution), the need for a public transportation system that is clean, efficient, beneficial and economical is ever more apparent. An excellent method to explore the optimization of such a system is through a computer model which runs scenarios to optimize unknown variables. The project, which was programmed in C++, using a complex set of classes, allows the user to model a diversity of public transportation systems, ranging from subways to light rail trains.

Upon opening, the program loads a world file, which defines the variables and scenarios the program will be running. Each class, ranging from the tracks to the people, is defined in this initial phase (in preparation for running). For example, within the world file, the trains are defined, including all data on speeds (acceleration, max velocity, deceleration and turning velocities). Upon initialization, the program loads the variables into the train classes, which will be used during the scenario.

The next phase of the program is the actual execution of the scenario. A master class (“the world class”) manages all the other classes, executing each iteration of the time-based model. The program’s iteration represents one second of real time, allowing for a much more precise model, but a higher running time than another modeling strategy. Within iteration, the program moves the trains and people, as well as records data.

Throughout the execution of the program, individual people are managed, tracking time at work and at home to position them at key stops during peak traffic hours. Each person’s satisfaction is measured throughout the program so as to determine the value of the public transportation system in this respect. Satisfaction is based on a variety of factors, including travel time and number of trains ridden. Using this method, a key aspect of the program is fulfilled, allowing it to determine the worth of the system in the running scenario.

When the program has run for a set period of time (usually the equivalent of one week, or 604,800 iterations), it will record all results to an output file, which allows the user to get detailed results and step-by-step data from the iterations. This allows the program operator to identify faults in the system, as well as cost information, peak hours and user satisfaction. The output file is a key part of the program, which gives the user the results of the scenarios and provides optimized variables that were specified when the program was run.

To test the program, three different scenarios were run based on the Albuquerque area, along with surrounding cities and suburbs. These samples were meant to determine the benefits of a public transportation system in the Albuquerque area.

The Code

The code behind the program was written in C++, which turned out to be inefficient because of the lack of access to graphical tools as well as advanced mathematical processing. In particular the poor design of memory management caused the 2D models to be inaccurate. Before the final presentation, a more accurate and user-friendly version will be completed in C#, allowing a graphical interface to be added. For more information on the code, see the “Class” appendix.

Results

Three key scenarios were created, which the program runs as a way to test the functionality of the code and create a real life application of the program. The scenarios are all based on the Albuquerque area. One scenario involves having a single-track running between Albuquerque and Santa Fe (an intercity route). The second scenario focuses on the central urban areas of Albuquerque (the university and downtown), while the third imitates a full city system, providing access to all key locations and stops on the routes. The results returned by the program provided some interesting facts, suggesting the Albuquerque area was not an optimal place for a public transportation system. Three key graphs are shown that summarize some of the results returned by the program.

The first graph shows cost breakdowns. Because exact costs were hard to come by, the numbers are relative and are used simply for comparison. Each route is broken down into “Initial cost” (installation), “Maintenance”, and “Revenue”. The suburban system, because of the high number of tracks, cars and stops has both a high initialcost and high maintenance fees, but little income due to complexity of installation and maintenance. The suburban scenario is not financially a sound choice for the city. The urban scenario, although low in initial cost, is also low in usage (because of low population density in urban Albuquerque), causing maintenance fees to be higher than revenue. Finally, the expensive intercity route has a higher profit yield, because of low maintenance fees. The best scenario with regard to finance is the intercity route, because all costs for building will be paid off by profits, without the high maintenance fees.

The second graph shows what turned out to be an important factor: distance from arrival stop to final destination. One of the key problems with the intercity route was the long commute after arrival in the destination city. Given the lack of a good bus network, the commute would be either by taxi or on foot. This was a significant factor in the failure of the intercity route, as many passengers would have no transportation method after arriving in Santa Fe (or vice versa). Until good innercity routes are established, the intercity scenario appears useless. Walking distance had little affect on the other two scenarios.

The third graph describes user satisfaction, a concept that, although hard to quantify, describes the recreation of the people utilizing the public transportation system. The numbers are based on a range of information, ranging from trip purpose, to route complexity, to number of people. The graph breaks the satisfaction into three categories: satisfied (the person will continue using it), moderately satisfied (the person may use it in certain situations) and finally not satisfied (the person will be unlikely to use it again). The percentages refer to the group of people who would even consider using public transportation, because there are a high number of people whose current dependence on and relationship with their automobile will cause them not to consider changing to public transportation. Interestingly, the satisfaction with the suburban scenario was greatly lowered because of its complexity, even though it offered high coverage. With numerous transfers and tracks required and a high initial cost, this scenario appears to be impractical.

For each scenario, there were many pros and cons, making the results of the program difficult to interpret.

Conclusions

According to the program, the best route for customer satisfaction is an urban center route, but because of the low usage that occurs as a result of the low population density, the scenario does not serve a large group of people. This scenario, therefore, is not economically sound and will not pay for itself, until the density of the urban center increases. The scenarios that did serve large areas with more people ran into satisfaction problems, because of complexity of routes and added travel distance/time.

The program's results, although highlighting the pros and cons of the scenarios, are not accurate and complete, due to the high complexity of the problem at hand. Even as the problem was simplified, new complexities appeared, making it apparent that the program is unable to embody the true complexities of a public transportation (and transportation) system. Especially when it comes to judging satisfaction, the simplified version falls short because it bases human emotions and responses on a few key factors. To truly measure the accuracy of the model, a far more complex program would have to be created. Further, modeling of issues such as traffic patterns and pedestrian movement, a substantial undertaking in itself, would have to be incorporated into the model.

Because of the many inaccuracies of the model, it represents progress toward our initial goal, but leaves much room for improvement. The current results seem only marginally relavent to existing situations because of the high level of inaccuracy. Future improvements will be made prior to the final presentation. This future work involves rewriting the core elements, focusing primarily on the two dimensional aspects.

In conclusion, the program provides an interesting model, which can be improved greatly, but which at present reflects little of the true applications of public transportation.

Recommendations

Many future improvements can be made to the program, as the true model involved is so complex. Currently, the following improvements will be made before the final presentation, in hopes of creating a more realistic modeling environment:

- **Programming language** – Using a visual studio conversion tool, all code will be imported into C#, allowing for a more usable edition of the program to be created, with the added support of easier 2D modeling.
- **Better Measurement of Success** – Currently, the program bases all satisfaction ratings on variables, which involve overly simplified artificial emotions, creating unrealistic satisfaction measurement. Because of the complexity of human satisfaction and response to public transportation, in order to adequately measure satisfaction, a much more realistic model must be created.
- **2D Model** – Currently, the two dimensional modeling layout is inefficient and inaccurate, requiring far too many formulas and values in the initial world field. This is a result of the poor variable management in C++. For this reason, the program will be exported to another language, allowing for better handling and management of a two dimensional world.

- **Traffic Modeling** – Although this improvement is too difficult to grapple with prior to the final presentation, it is an important aspect to consider and eventually add. Because of the complexity of driving (changing lanes, signals, etc), traffic modeling was not incorporated into the initial program. This exclusion out diminishes any connection with reality, since traffic patterns are an integral component of transportation systems.

Many improvements can be made to the program, helping to ground it in reality. A few of these will be tackled before Awards Day, but many are too complex to even consider adding at this time.

Acknowledgements

Challenge Team 20 would like to thank the following people for their contributions to our project:

- Debra Loftin for her help and support as our sponsoring teacher, and putting up with both Nathans. We also thank her for sponsoring the AIS Challenge at Bosque School. Without her, there would be no reason to do the AIS Challenge.
- Penny Perkins for supporting us by providing comments, transportation, and editing assistance.
- Joseph and Ann Merrick for their support and transportation.
- Ian Ford for his input to our project; his input was especially valuable as he sits on the Future Transit Board.
- Andrew Wooden and The Bosque School Faculty and for their support and leniency when it was required of us to miss class and their support in using school resources.
- Greg Scantlen for his computer support and allowing us to use school technology resources for our project.
- Brad Key, Maximo Lazo, and Thomas Laub for their helpful comments and suggestions, which aided us in further progress.
- The wonderful people at Consult for their support and help in choosing a programming language and help with our Abstract.

References

American Public Transportation Society. January 9, 2003 <http://www.apta.com/>.

American Public Transportation Society. APTA Statistics. July 23, 2002 <http://www.apta.com/stats/>.

Liberty, Jesse. Teach Yourself C++ in 21 days. New York: Sams, USA (1999).

LTAP Resource Page. March 9, 2002 <http://www.ltap.psu.edu/aboutltap.cfm>.

McNichol, Dan. The Big Dig. Boston: Silver Lining Books (November 21, 2000).

Microsoft Corporation. Microsoft .NET Home Page. March 4, 2002 <http://msdn.microsoft.com/>.

Newman, Peter. Sustainability and Cities: Overcoming Automobile Dependence. London: Island Press (February, 1999).

Ohio LTAP. RoadWare Modeling Software. June 30, 2001 <http://www.ohioltap.org/ltap/LTAPSoftware.htm>.

Southworth, Michael. Streets and the Shaping of Towns and Cities. New York: McGraw-Hill Professional (August 22, 1996).

United States Government. Bureau of Transportation Statistics. March 12, 2003 <http://www.bts.gov/>.

United States Government. Department of Transportation. February 3, 2003 <http://www.dot.gov/>.

United States Government. National Transportation Library. <http://www.bts.gov/ntl/>.

University of Berkeley. “FREQ12 Simulation.” Computing Media. September 6, 2000 <http://www.its.berkeley.edu/computing/software/freq.html>.

University of Berkeley. “Public Transportation System.” Computing Media. <http://www.path.berkeley.edu/PATH/Publications/Media/FactSheet/PublicTransit.pdf>.

Appendix A: Classes

- Person
  - Functions
    - Is Person Available – Is the person currently waiting
    - Start Route – Initiates a trip
    - Board Train
    - Transfer Trains
    - End Route
    - Run Iteration
    - Ready to Board – Check if the person is ready to board
  - Variables
    - Mood
    - Personality
    - Location – Stop #, train # or 0
    - Status – Waiting, on train, station
    - Destination – ID of the destination
    - Start – ID of the starting stop
    - Transfer Count – Number of transfers for the trip
    - Transfers – An array of transfer classes
    - Current Transfer – Location in the transfer array
    - Exit Stop – The stop the person will exit the train
    - Travel Start – Time that the trip starts
    - Travel Stop – Time that the trip ends
    - Traveling With – Number of people traveling with
    - Delay – Count down timer to wait at current stop
    - Waiting to Board – The track which the person is waiting to board
    - Ride Rating – The rating of the current trip
- Train
  - Functions
    - Is Room – Sees if the train has room for more people
    - Run Iteration – Moves the train, updating location, speed, etc
    - Update Speed – Recalculates speed based on location
  - Variables
    - Location – Track segment, stop number or 0
    - Speed
    - Speed Status – Increasing, decreasing, steady or turning
    - Turn counter – Used for adjusting time for curves
    - Status – Moving, resetting, waiting
    - Number of passengers – The number of *actual* people on the train
    - Array of passengers – Contains an array of all persons on the train
    - Remaining Dist – Remaining distance on track segment
- Track

  - Functions
    - Get Segment
    - Get Next Segment
    - Is Point
    - Is Stop
    - Get Stop
    - Get Next Stop
    - Get Next Point
  - Variables
    - Lengths – Length of track segments (array)
    - Number of Segments
    - Array of Segments – The track segments that compose the 2D model
    - Number of Trains
    - Array of Trains
    - Number of Stops
    - Array of Stops
    - Number of points – The number of important track points
    - Array of Points – Array of important track points
- Stop
  - Functions
    - Output statistics – Outputs the actual statistics stored in unlisted variables
    - Run iteration – Moves people around area and collects statistics
  - Variables
    - Tracks
    - People waiting
    - Location – Point of the actual stop
- World
  - Functions
    - Load world file
    - Run iteration
    - Output results
  - Variables
    - Array of stops
    - Array of tracks
    - Array of trains
    - Array of people
- Point
  - Functions
    - setPoint
    - getPoint – Returns an array with the two variables
    - getPointX – Returns X
    - getPointY – Returns Y
  - Variables
    - X – The X coordinate

        - Y – The Y coordinate
- Transfer
    - Functions
        - setTransfer – Set up the transfer variable
        - getStop – Get the stop number
        - getCurrent – Get the current track
        - getNext – Get the next track
    - Variables
        - StopID – The stop number
        - TrackIDa – The first track
        - TrackIDb – The second track

Appendix B: Program Walkthrough (Flow chart)

1. Load World File
    a) Create the people
        - Define personalities and moods
        - Set locations
        - Set status
        - Set travel load
    b) Create 2D map
        - Load tracks
        - Create segments and points
        - Insert Stops
    c) Load train data
        - Create trains
        - Set speeds and capacities
2. Begin program
    a) Calculate costs
    b) Run iteration
        - Train movement
            - Relocate train
                - Detect a stop
                - Detect a turn
            - Update speed
                - Determine acceleration/deceleration
                - Measure turns
        - Stops – Upon train arrival
            - Check for transfer exits
                - Recalculate satisfaction
            - Check for route ends
                - Measure walking distance
                - Measure satisfaction
            - Set arriving persons wait times

- Add new people based on times set in world file
    - Update mood
    - Update location data
- Collect exiting data
- Advance iteration and begin again

3. Output collected statistics and use them to arrive at generalized conclusions on scenario

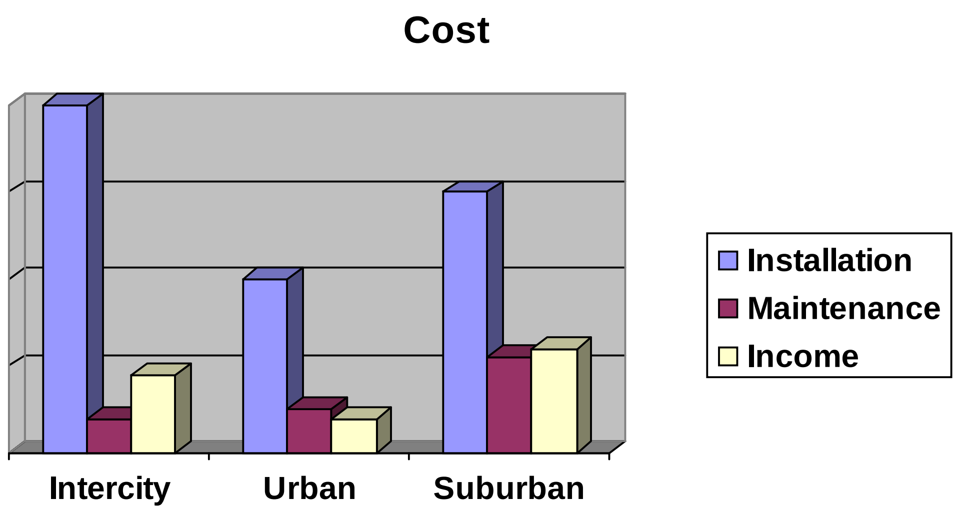
Appendix C: Example World File

```
# Example World File

[World]
Population=500

[Tracks]
Tracks=2

[Track0]
TotalDistance=20
Segments=3
Segment0=4
Segment1=2
Segment2=6

Point0=0,0
Point1=0,4
Point2=2,4
Point3=8,4

[Track1]
TotalDistance=20
Segments=3
Segment0=6
Segment1=2
Segment2=4

Point0=8,4
Point1=2,4
Point2=0,4
Point3=0,0

[Stops]
Stops=3
```

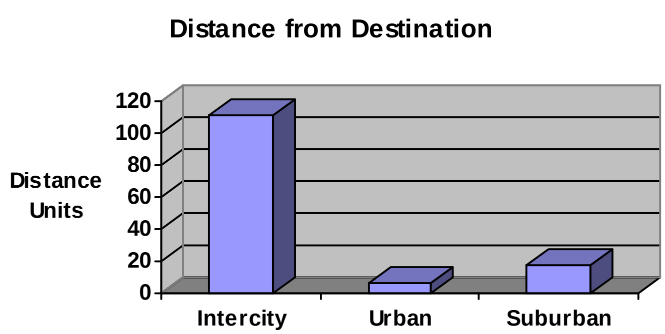
```
[Stop0]
Tracks=0
TimeForTrain=12
Location=0,0

[Stop1]
Tracks=0,1
TimeForTrain=8
Location=2,4

[Stop2]
Tracks=0,1
TimeForTrain=12
Location=8,4

[Trains]
Train=2

[Train0]
Speed=80
#MPH
Acceleration=4
# Per second
Decceleration=3
Capacity=50
TurnPause=1

[TimeStop0]
Time0:00=5
Time0:30=3
Time1:00=2
Time1:30=2
Time2:00=2
Time2:30=2
Time3:00=2
Time3:30=2
Time4:00=6
Time4:30=6
Time5:00=8
Time5:30=9
Time6:00=12
Time6:30=12
Time7:00=18
Time7:30=22
Time8:00=24
Time8:30=26
Time9:00=18
```

```
Time9:30=14
Time10:00=10
Time10:30=6
Time11:00=9
Time11:30=18
Time12:00=20
Time12:30=20
Time13:00=18
Time13:30=9
Time14:00=6
Time14:30=4
Time15:00=6
Time15:30=8
Time16:00=8
Time16:30=12
Time17:00=18
Time17:30=24
Time18:00=28
Time18:30=14
Time19:00=8
Time19:30=6
Time20:00=6
Time20:30=9
Time21:00=9
Time21:30=6
Time22:00=6
Time22:30=5
Time23:00=4
Time23:30=5

# Extra Cut Off
```

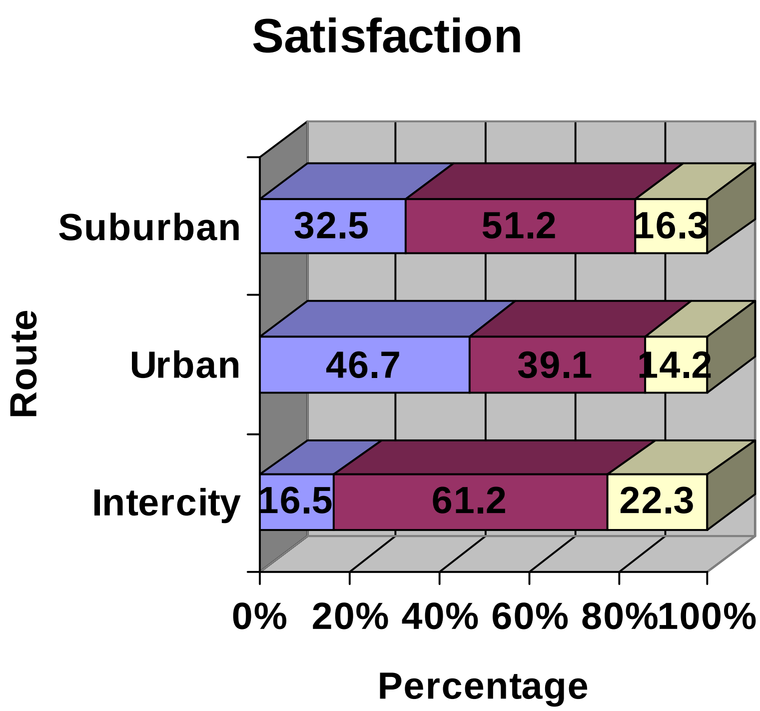
Appendix D: Graphs



Graph 1 – Cost



Graph 2 – Distance from Destination



	Intercity	Urban	Suburban
Not satisified	22.3	14.2	16.3
Moderately Satisified	61.2	39.1	51.2
Satisified	16.5	46.7	32.5

Graph 3 – Satisfaction