ANALYSIS OF AIRPORT APM SYSTEMS
USING A COMPUTER SIMULATION MODEL

by
Yi-Dar Lin

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of
Master of Science
in
Civil Engineering

APPROVED:

Antonio A. Trani, Ph.D., Chairman

Donald R. Drew, Ph.D.  Richard D. Walker, Ph.D.

September, 1995
Blacksburg, Virginia
ANALYSIS OF AIRPORT APM SYSTEMS
USING A COMPUTER SIMULATION MODEL

by
Yi-Dar Lin
Dr. A.A. Trani, Chairman
Civil Engineering

(ABSTRACT)

Automated People Movers (APM) have become an attractive solution to problems associated with the concept of airside/landside airport development. This technology, if used effectively, could provide an acceptable level of service to airport users and a good operating efficiency to airports and airlines.

The main objective of this research is to develop a computer simulation model to simplify the operational analysis of airport APM systems and to assess their level of service characteristics. The model simulates the movement of individual passengers and vehicles in the system network, and provides a tool for planners and designers: (1) to determine the sensitivity of system performance for a range of APM design parameters, (2) to examine the flexibility of an APM system under given operational policy and network configurations, and (3) to estimate the APM vehicle energy consumption based on network constraints and system characteristics.

The model is a discrete-event simulation model developed using EXTEND (Version 3.0.2, © Imagine That, Inc., 1994) software. With the powerful features of animation, graphics, and hierarchical modeling, the EXTEND libraries provide the capability to easily model alternative service concepts such as shuttles, loops, and single or double routes.
ACKNOWLEDGMENTS

I sincerely wish to express my gratitude to my advisor Dr. Trani for his full support, expert guidance, understanding and encouragement throughout the course of this research. Also, I am greatly indebted to him for his critical review of the manuscript of my thesis.

I would like thank my committee members, Dr. Drew and Dr. Walker, for serving on my committee. Especially, I thank Dr. Drew for providing me with excellent course instruction during my graduate years at Virginia Tech.

It is my greatest pleasure to dedicate this thesis to my parents whose dream it was to see me accomplish this and whose love helped me fulfill it. I would also like to thank my sister, brother-in-law, brother, and sister-in-law for their unconditional support and endless affection throughout my life.

Finally, I would like to thank the many friends who encouraged me and made my live in the United States an enjoyable one.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. <strong>Introduction</strong> ................................................................. 1</td>
<td></td>
</tr>
<tr>
<td>1.1 Background ................................................................. 1</td>
<td></td>
</tr>
<tr>
<td>1.2 Airport Systems ............................................................. 2</td>
<td></td>
</tr>
<tr>
<td>1.3 Airline Operations .......................................................... 4</td>
<td></td>
</tr>
<tr>
<td>1.4 Research Scope, Objective, and Approach ................................ 5</td>
<td></td>
</tr>
<tr>
<td>2. <strong>Literature Review</strong> ............................................................ 7</td>
<td></td>
</tr>
<tr>
<td>2.1 Introduction ................................................................. 7</td>
<td></td>
</tr>
<tr>
<td>2.2 The APM Planning and Design Process ...................................... 7</td>
<td></td>
</tr>
<tr>
<td>2.3 Requirements Analysis and Concepts Development ......................... 11</td>
<td></td>
</tr>
<tr>
<td>2.3.1 Level of Service Analysis ................................................. 11</td>
<td></td>
</tr>
<tr>
<td>2.3.2 APM Demand Analysis ...................................................... 13</td>
<td></td>
</tr>
<tr>
<td>2.3.3 Capacity Analysis .......................................................... 14</td>
<td></td>
</tr>
<tr>
<td>2.3.3.1 Guideway Capacity ...................................................... 15</td>
<td></td>
</tr>
<tr>
<td>2.3.3.2 Station Capacity ......................................................... 20</td>
<td></td>
</tr>
<tr>
<td>2.3.4 Flow Analysis ............................................................... 22</td>
<td></td>
</tr>
<tr>
<td>2.3.5 Description of Selected Simulation Approaches .......................... 28</td>
<td></td>
</tr>
<tr>
<td>2.3.6 Energy Consumption Analysis ............................................ 31</td>
<td></td>
</tr>
<tr>
<td>3. <strong>Methodology</strong> ................................................................. 34</td>
<td></td>
</tr>
<tr>
<td>3.1 Introduction ................................................................. 34</td>
<td></td>
</tr>
<tr>
<td>3.2 Systems, Models, and Simulation ........................................... 34</td>
<td></td>
</tr>
</tbody>
</table>
Chapter 3.3 Approach in a Simulation Study ........................................... 36
3.4 Description of EXTEND ......................................................... 39

4. Model Description .................................................................. 42
   4.1 Introduction .......................................................................... 42
   4.2 Model Specifications .......................................................... 42
       4.2.1 System Components ..................................................... 42
       4.2.2 System Activities and Associated Events ...................... 43
   4.3 Model Assumptions ............................................................. 43
   4.4 Simulation Model Logic ....................................................... 51
       4.4.1 APM Station Model ..................................................... 51
       4.4.2 APM Guideway Model ............................................... 54
   4.5 Energy Consumption Model ............................................... 58
   4.6 Output of the Model ........................................................... 59
   4.7 Block Reference for the APM Libraries ............................... 61
       4.7.1 Alphabetical Block Reference
           for the APM Station Library ......................................... 63
       4.7.2 Alphabetical Block Reference
           for the APM guideway Library ................................. 69
       4.7.3 Alphabetical Block Reference
           for the New Utility Library .................................... 75

5. Model Application and Analysis ............................................. 115
   5.1 Introduction ................................................................. 115

TABLE OF CONTENTS
Chapter | Page
---|---
5.2 Model Application | 115
5.2.1 Model Assumptions | 117
5.2.2 Base Year Scenario | 118
5.2.3 Horizon Year Scenario | 119
5.3 Model Results | 119

6. Conclusions and Recommendations | 136
6.1 Introduction | 136
6.2 Conclusions | 136
6.3 Recommendations | 138

Bibliography | 138

Appendix A. Generic Library Blocks Used In The Model | 143
Appendix B. Discrete Event Library Blocks Used In The Model | 146
Appendix C. Plotter and Animation Library Blocks Used In The Model | 155
Appendix D. Energy Consumption Model | 157
# LIST of FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>APM Planning and Design Process</td>
<td>9</td>
</tr>
<tr>
<td>2.</td>
<td>Example of the Passenger Flow Pattern in an APM Station Model</td>
<td>53</td>
</tr>
<tr>
<td>3.</td>
<td>The Basic APM System Configurations</td>
<td>56</td>
</tr>
<tr>
<td>4.</td>
<td>Alphabetical List of the APM Station Library</td>
<td>62</td>
</tr>
<tr>
<td>5.</td>
<td>Alphabetical List of the APM Guideway Library</td>
<td>68</td>
</tr>
<tr>
<td>6.</td>
<td>Alphabetical List of the APM New Utility Library</td>
<td>74</td>
</tr>
<tr>
<td>7.</td>
<td>Structure of the Circulation Area/Passageway Block</td>
<td>83</td>
</tr>
<tr>
<td>8.</td>
<td>Structure of the Elevator Block</td>
<td>84</td>
</tr>
<tr>
<td>9.</td>
<td>Structure of the Entrance/Exit Block</td>
<td>85</td>
</tr>
<tr>
<td>10.</td>
<td>Structure of the Escalator Block</td>
<td>86</td>
</tr>
<tr>
<td>11.</td>
<td>Structure of the Pax-Assignment Block</td>
<td>87</td>
</tr>
<tr>
<td>12.</td>
<td>Structure of the Pax-Schedule (Method 1) Block</td>
<td>88</td>
</tr>
<tr>
<td>13.</td>
<td>Structure of the Pax-Schedule (Method 2) Block</td>
<td>89</td>
</tr>
<tr>
<td>14.</td>
<td>Structure of the Pax-Schedule (Method 3) Block</td>
<td>90</td>
</tr>
<tr>
<td>15.</td>
<td>Structure of the Platform Block</td>
<td>91</td>
</tr>
<tr>
<td>16.</td>
<td>Structure of the Sidewalk Block</td>
<td>92</td>
</tr>
<tr>
<td>17.</td>
<td>Structure of the Stairway Block</td>
<td>93</td>
</tr>
<tr>
<td>18.</td>
<td>Structure of the Transit Unit Block</td>
<td>94</td>
</tr>
<tr>
<td>19.</td>
<td>Structure of the 2-Way Switch (1) Block</td>
<td>95</td>
</tr>
<tr>
<td>20.</td>
<td>Structure of the 2-Way Switch (2) Block</td>
<td>96</td>
</tr>
<tr>
<td>21.</td>
<td>Structure of the Merge/Diverge (1) Block</td>
<td>97</td>
</tr>
<tr>
<td>22.</td>
<td>Structure of the Merge/Diverge (2) Block</td>
<td>98</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>23</td>
<td>Structure of the Merge/Diverge (3) Block</td>
<td>99</td>
</tr>
<tr>
<td>24</td>
<td>Structure of the Merge/Diverge (4) Block</td>
<td>100</td>
</tr>
<tr>
<td>25</td>
<td>Structure of the Pinched Loop (1) Block</td>
<td>101</td>
</tr>
<tr>
<td>26</td>
<td>Structure of the Pinched Loop (2) Block</td>
<td>102</td>
</tr>
<tr>
<td>27</td>
<td>Structure of the Single-Lane (1), (2), (3) or (4) Block</td>
<td>103</td>
</tr>
<tr>
<td>28</td>
<td>Structure of the Single-Lane (5) Block</td>
<td>104</td>
</tr>
<tr>
<td>29</td>
<td>Structure of the Single-Lane (6) Block</td>
<td>105</td>
</tr>
<tr>
<td>30</td>
<td>Structure of the Turnaround (1) Block</td>
<td>106</td>
</tr>
<tr>
<td>31</td>
<td>Structure of the Turnaround (2) Block</td>
<td>107</td>
</tr>
<tr>
<td>32</td>
<td>Structure of the Veh-Schedule Block</td>
<td>108</td>
</tr>
<tr>
<td>33</td>
<td>Structure of the APM Plotter Block</td>
<td>109</td>
</tr>
<tr>
<td>34</td>
<td>Structure of the Boarding/Debarking Block</td>
<td>110</td>
</tr>
<tr>
<td>35</td>
<td>Structure of the Distributor (1) Block</td>
<td>111</td>
</tr>
<tr>
<td>36</td>
<td>Structure of the Distributor (2) Block</td>
<td>112</td>
</tr>
<tr>
<td>37</td>
<td>Structure of the General Information Block</td>
<td>113</td>
</tr>
<tr>
<td>38</td>
<td>Structure of the Guideway Maneuver Block</td>
<td>114</td>
</tr>
<tr>
<td>39</td>
<td>ATL APM System Configuration</td>
<td>116</td>
</tr>
<tr>
<td>40</td>
<td>ATL Airport APM Guideway Model</td>
<td>120</td>
</tr>
<tr>
<td>41</td>
<td>ATL Airport APM Station Models</td>
<td>121</td>
</tr>
<tr>
<td>42</td>
<td>Travel Distance vs. Simulation Time Profile for TU No.4</td>
<td>125</td>
</tr>
<tr>
<td>43</td>
<td>Travel Distance vs. Simulation Time Profile for TU No.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Traveling from Ticketing Station to Concourse A Station</td>
<td>125</td>
</tr>
<tr>
<td>44</td>
<td>Acceleration vs. Simulation Time Profile for TU No.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Traveling from Ticketing Station to Concourse A Station</td>
<td>126</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>45.</td>
<td>Acceleration vs. Distance Profile for TU No.4</td>
<td>126</td>
</tr>
<tr>
<td></td>
<td>Traveling from Ticketing Station to Concourse A Station</td>
<td></td>
</tr>
<tr>
<td>46.</td>
<td>Velocity vs. Simulation Time Profile for TU No.4</td>
<td>127</td>
</tr>
<tr>
<td></td>
<td>Traveling from Ticketing Station to Concourse A Station</td>
<td></td>
</tr>
<tr>
<td>47.</td>
<td>Velocity vs. Distance Profile for TU No.4</td>
<td>127</td>
</tr>
<tr>
<td></td>
<td>Traveling from Ticketing Station to Concourse A Station</td>
<td></td>
</tr>
<tr>
<td>48.</td>
<td>Power Required for TU No.4</td>
<td>128</td>
</tr>
<tr>
<td></td>
<td>Traveling from Ticketing Station to Concourse A Station</td>
<td></td>
</tr>
<tr>
<td>49.</td>
<td>Energy Consumption for TU No.4</td>
<td>128</td>
</tr>
<tr>
<td></td>
<td>Traveling from Ticketing Station to Concourse A Station</td>
<td></td>
</tr>
<tr>
<td>50.</td>
<td>Energy Consumption for TU No.4</td>
<td>129</td>
</tr>
<tr>
<td></td>
<td>/////////////////////////////////////////////////////////////////////////////</td>
<td></td>
</tr>
<tr>
<td>51.</td>
<td>Load Factor for TU No.4 Traveling from Ticketing Station</td>
<td>129</td>
</tr>
<tr>
<td></td>
<td>to Concourse A Station (Base/Horizon Year Scenario)</td>
<td></td>
</tr>
<tr>
<td>52.</td>
<td>Occupancy for TU No.4 Traveling from Ticketing Station</td>
<td>130</td>
</tr>
<tr>
<td></td>
<td>to Concourse A Station (Base/Horizon Year Scenario)</td>
<td></td>
</tr>
<tr>
<td>53.</td>
<td>LOS in TU No.4 (Base Year Scenario)</td>
<td>130</td>
</tr>
<tr>
<td></td>
<td>/////////////////////////////////////////////////////////////////////////////</td>
<td></td>
</tr>
<tr>
<td>54.</td>
<td>Average Waiting Time on the Platform at North</td>
<td>131</td>
</tr>
<tr>
<td></td>
<td>Concourse A Station (Base/Horizon Year Scenario)</td>
<td></td>
</tr>
<tr>
<td>55.</td>
<td>Passenger Queuing on the Platform at North</td>
<td>131</td>
</tr>
<tr>
<td></td>
<td>Concourse A Station (Base/Horizon Year Scenario)</td>
<td></td>
</tr>
<tr>
<td>56.</td>
<td>LOS on the Platform at North</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td>Concourse A Station (Base/Horizon Year Scenario)</td>
<td></td>
</tr>
<tr>
<td>57.</td>
<td>Number of Passengers on the Platform at North</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td>Concourse A Station (Base/Horizon Year Scenario)</td>
<td></td>
</tr>
</tbody>
</table>

**LIST OF FIGURES**
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>58.</td>
<td>LOS on the Platform at North Concourse A Station</td>
<td>133</td>
</tr>
<tr>
<td></td>
<td>(Base Year Scenario)</td>
<td></td>
</tr>
<tr>
<td>59.</td>
<td>Number of Passengers Entering/Leaving the Platform at North Concourse A Station</td>
<td>133</td>
</tr>
<tr>
<td>60.</td>
<td>LOS on the Escalator at Concourse A Station</td>
<td>134</td>
</tr>
<tr>
<td></td>
<td>(Base/Horizon Year Scenario)</td>
<td></td>
</tr>
<tr>
<td>61.</td>
<td>LOS on the Stairway at Concourse A Station</td>
<td>134</td>
</tr>
<tr>
<td></td>
<td>(Base/Horizon Year Scenario)</td>
<td></td>
</tr>
<tr>
<td>62.</td>
<td>Number of Passenger Arriving at Concourse A Station by APM</td>
<td>135</td>
</tr>
<tr>
<td></td>
<td>(Base/Horizon Year Scenario)</td>
<td></td>
</tr>
<tr>
<td>63.</td>
<td>Number of Passenger Taking the Elevator at Concourse A Station</td>
<td>135</td>
</tr>
<tr>
<td></td>
<td>(Base/Horizon Year Scenario)</td>
<td></td>
</tr>
</tbody>
</table>
LIST of TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Airport APM Systems</td>
<td>3</td>
</tr>
<tr>
<td>2. APM Simulation Variables Related to Ridership, Cost, and Performance Specifications</td>
<td>10</td>
</tr>
<tr>
<td>3. LOS Standards for Vehicles, Waiting Area, and Walking Area</td>
<td>12</td>
</tr>
<tr>
<td>4. Safety Regimes of Transit Operation: Combinations of Deceleration Types for Successive Transit Units</td>
<td>18</td>
</tr>
<tr>
<td>5. Factors to Influence Transit Capacity</td>
<td>23</td>
</tr>
<tr>
<td>6. Values for Use in the Davis' Equation</td>
<td>33</td>
</tr>
<tr>
<td>7. Entities and Their Attributes of an APM System</td>
<td>45</td>
</tr>
<tr>
<td>8. Possible Activities in the APM Simulation</td>
<td>49</td>
</tr>
<tr>
<td>9. Possible Events Associated with System Activities in the APM Simulation</td>
<td>50</td>
</tr>
<tr>
<td>10. Flight Schedule during Busy Hours at ATL Airport</td>
<td>123</td>
</tr>
</tbody>
</table>
1. INTRODUCTION

1.1 Background

In the United States, the commercial aviation industry has matured since the end of World War II. The remarkable growth in air travel during the past years has not been matched by a similar growth in airport capacity. The number of domestic and international revenue passenger miles (RPMs) of U.S. major airlines grew from 157.9 billion RPMs in 1973 to 212.5 billion RPMs in 1978. By 1988, after a decade of the Airline Deregulation Act of 1978, the number of domestic and international RPMs has reached 420.4 billion. The Federal Aviation Administration (FAA) forecasts that the total number of RPMs will reach 777.9 billion in 2001 [FAA, 1978 and 1990]. The growth trend during the past years in United States air transportation will continue into the 21st century.

As airport terminal facilities are expended to accommodate such rapid growth, airports will require more reliable passenger transit technologies to provide an acceptable level of service to airport users, and a good operating efficiency to airports and airlines. Without passenger terminal technologies, passengers would probably incur intolerable walking distances and aircraft-to-aircraft transfer times at current larger airports.

Since 1970, there has been increasing interest within the United States in the automated people mover (APM) systems as passenger transit technologies. APM systems are fully automated, driverless vehicles that operate on a fixed guideway through an exclusive right-of-way. Such systems were initially installed for amusement center applications such as Walt Disney World in 1974 and have successfully been developed in controlled environments such as major metropolitan area and airports. In April of 1986, Miami became the first city in the United States to have an integrated automated people mover (APM) system and the second downtown people mover system has been operated in Detroit since 1987. In addition, the first installation of an APM system at a major U.S.
airport and in the world was at Tampa International Airport in 1971. Today, there are about 13 airports with APM systems. Table 1 shows the current status of APM systems developed in U.S. major airports.

1.2 Airport Systems

Several passenger transit systems are used today at U.S. airports to improve passenger mobility. They are moving sidewalks, courtesy carts, buses, and APMs. Traditionally, APM systems have been used to reduce the impact of passengers walking between a main terminal, in which ticketing and baggage claim occurs, and aircraft gates often located in airside satellite terminals or piers. The systems have also been used to connect main terminals and airport landside facilities such as remote parking areas and car rental facilities. Due to the restriction of land uses around airports, additional terminals were built instead of expanding an individual terminal, and more airport planners began to use APM systems to link these various terminals. Since passage of the Airline Deregulation Act of 1978, the hub-and-spoke service pattern has increased peak demands of intra-line transfer passengers and greater numbers of aircraft gates during connections. The expected gate-to-gate as well as gate-to-entrance (or exit) distances by transferring passengers become longer due to complex airport configurations. As a result, APM systems instead of typical means of conveyance such as walking and moving sidewalks have been applied to provide transferring passengers an improved level of service to meet the trip time requirements for transfers.

More recently, the on-airport and off-airport APM systems have been applied to support further integration of the airport with the surrounding city or community areas for transportation needs. For example, APM systems are being used as the interface transportation mode between airports and city rail transit systems to improve accessibility to airports for nonautomotive travel and provide the necessary capacity while meeting the level 1. INTRODUCTION
Table 1. Airport APM Systems [Austin, 1993]

<table>
<thead>
<tr>
<th>Airport</th>
<th>Start of service</th>
<th>Approximate length of guideway (mi)</th>
<th>Number of stations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hartsfield International, Atlanta Extension</td>
<td>1980</td>
<td>2.4</td>
<td>10</td>
</tr>
<tr>
<td>O'Hare International, Chicago</td>
<td>1993</td>
<td>2.7</td>
<td>5</td>
</tr>
<tr>
<td>Cincinnati International</td>
<td>1994</td>
<td>0.25</td>
<td>3</td>
</tr>
<tr>
<td>Dallas-Fort Worth International Modifications</td>
<td>1974</td>
<td>13</td>
<td>28</td>
</tr>
<tr>
<td>New Denver International</td>
<td>1993</td>
<td>2.4</td>
<td>7</td>
</tr>
<tr>
<td>Houston Inter-Continental Extension</td>
<td>1972</td>
<td>1.4</td>
<td>9</td>
</tr>
<tr>
<td>Newark International</td>
<td>1990</td>
<td>0.19</td>
<td>2</td>
</tr>
<tr>
<td>McCarran International, Las Vegas</td>
<td>1985</td>
<td>0.50</td>
<td>2</td>
</tr>
<tr>
<td>Miami International</td>
<td>1980</td>
<td>0.50</td>
<td>2</td>
</tr>
<tr>
<td>Orlando International</td>
<td>1981</td>
<td>1.5</td>
<td>4</td>
</tr>
<tr>
<td>Extension</td>
<td>1989</td>
<td>0.70</td>
<td>2</td>
</tr>
<tr>
<td>Pittsburgh International</td>
<td>1992</td>
<td>0.93</td>
<td>2</td>
</tr>
<tr>
<td>Seattle-Tacoma International</td>
<td>1973</td>
<td>1.7</td>
<td>8</td>
</tr>
<tr>
<td>Tampa International</td>
<td>1971</td>
<td>1.4</td>
<td>8</td>
</tr>
<tr>
<td>Extension</td>
<td>1987</td>
<td>0.50</td>
<td>2</td>
</tr>
<tr>
<td>Extension</td>
<td>1991</td>
<td>0.60</td>
<td>6</td>
</tr>
<tr>
<td>Honolulu International</td>
<td>2000</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

1. INTRODUCTION
of service requirement of the airport in a cost effective manner. Moreover, possible expansion of APM systems at airports has been developed to link airport-related activity centers, such as remote parking areas, hotels, rental car facilities, air cargo facilities and office complexes, and even other airports. One of the obvious advantages in developing APM systems is that a decentralized airport allows more space for aircraft and support service vehicles to maneuver around apron ways and to gates.

1.3 Airline Operations

Since the introduction of wide-bodied jet aircraft (such as Boeing 747s, L-1011s, and DC-10s) into the fleets in the 1970s, airlines have generated greater load factors. After deregulation, many commercial airlines have changed their operations from conventional multiple “point-to-point” service to “hub-and-spoke” operation. With hub and spoke, a number of origins are routed via a central hub, where passengers can transfer between aircraft to a number of points served by the hub. Airlines prefer to concentrate the flow of passengers at a single hub airport within a metropolitan area to maximize transfer possibilities with the minimum number of flights. As a result, hub-and-spoke operations would place large volumes of transit and transfer demands on hub airport facilities.

Hub airports, besides serving origin-destination passengers, are used as transfer nodes where as many as 60 to 70 percent of the passengers only make connections from one flight to another. Aircraft from spoke origins arrive at a central hub, passengers are exchanged, and the aircraft then depart for spoke destinations. Within the tight time constraints in the hub transfer operation, the airlines are interested in some passenger transit systems for completing the connection process as quickly as possible to maximize the productivity of their aircraft fleets.

At most international airports, international flights operate from the international terminal and domestic flights operate from the domestic terminal. Therefore, passengers
transferring from domestic to international, or from international to domestic, must confront the great distances and times associated with terminals. Moreover, many hub airports that also serve large numbers of origin-destination passenger have large-scale terminal facilities with long walking distances. However, to accommodate such tremendous demands with particular airline operations, the APM should provide a high level of service to passengers, visitors, and airline and airport employees.

1.4 Research Scope, Objective, and Approach

Based on the above discussion, it could be concluded that there are three critical reasons for the development of APM technology to aid passenger mobility between airport facilities. They are: (a) continued highly growth in all categories of air travel in the future, (b) airline hubbing, which requires the transfer of large numbers of connecting passengers over long terminal distances in a short time, and (c) improvement of terminal circulation or landside access capacity. The APM systems function as an airport internal transit mode and an interface transit mode beyond airport. In airport capacity analysis, the APM systems improve the landside capacity of individual airport components such as the ground access, terminal, parking, and so on.

The objective of this research is to create a flexible simulation model that can be used to analyze the operation of APM systems at airports and assess the difference in their varied service characteristics. The model could be reasonably straightforward process to modify this simulation model to explore any realistic APM system design changes. In addition, this research study attempts to develop a simplified energy consumption model to estimate the energy cost of APM systems.

In short, the model simulates an APM system to provide a tool for planners and designers:

1. To model individual passenger and APM vehicle movement at airports.

1. Introduction
2. To determine the sensitivity of system performance to the range of APM design parameters (such as capacity, safe headway, and speed).

3. To estimate the APM vehicle energy consumption based on network constraints, vehicle characteristics and system characteristics.

4. To examine the flexibility of an APM system under given management algorithms, service policies, demand and network configurations.

As indicated before, APM systems must be planned as one part of major transportation system in airport design. The analysis and determination of the optimal system involves many variables of the APM system that can be analyzed through simulation. This process could be time consuming, and the complexity of network could preclude the use of analytical solutions. Therefore, a direct computer simulation appears to be a good tool for analysis.

This research presents the methodology and approach to build a computer simulation model using a simulation software, called EXTEND (Version 3.0.2, © Imagine That, Inc., 1994). This special purpose computer software provides a means to conveniently represent processes occurring in real-world APM systems and precisely simulate the operations of systems using a personal computer.

1. INTRODUCTION
2. LITERATURE REVIEW

2.1 Introduction

This literature review is intended to present some background of past and current research on the design and analysis of automated people mover (APM) systems at airports. This review briefly describes some fundamentals of APM system characteristics including APM planning and design process, level of service analysis, capacity analysis, flow analysis, energy consumption analysis. One selected simulation model is also reviewed at the end of this chapter.

2.2 The APM Planning and Design Process

The APM planning and design process can be simply divided into three phases:

1. Planning process
2. Design process
3. Implementation

The purpose of the planning process is to identify the problem, establish parameters and examine the feasibility of APM systems in comparison with other modes (i.e., moving sidewalks, courtesy carts, buses, and others) to reduce walking distance and travel time. In this process, it is necessary to understand the purposes that the APM system fulfills and the potential demand for ridership resulting from a level of service. After the baseline demand and travel desires have been identified, it is possible to establish a baseline network, such as station locations, guideway alignments, service requirements, and the resulting physical requirements for the APM system (i.e., vehicles, guideway, stations, etc.).

Figure 1 illustrates this concept of the APM simulation model which provides a framework to link the planning and design processes [Dooley, 1980]. Once the functions have been clearly identified in the planning process, it is possible to focus on route

2. LITERATURE REVIEW
alignment constraints and required station location options based on the layout of airport facilities. The design process defines the detailed system characteristics that will provide the level of service assumed in the demand estimation and planning process. Hence, the system designer uses the guideway layout including station locations and the interstation spacing and the baseline station-to-station demand matrix as given to identify performance requirements for APM systems. These include variables such as speed, capacity, safety, reliability, and maintainability. In the design process, it is necessary to consider system service characteristics used in the planning process, such as headways and travel times. That is, the system designer incorporates network constraints and demand profile assumptions into the scenario representation. Then, sets of system operating and hardware characteristics that satisfy the baseline service characteristics are defined.

Table 2 shows the system variables that can be manipulated by the APM simulation model and the corresponding service characteristic variables that are model outputs or can be derived from model outputs [Dooley, 1980]. Sensitivity analysis is used to help determine the level of model detail and determine which variables will have the greatest impact on the desired system performance. This analysis is performed in the areas shown in Figure 1. These include network constraints, demand projections, alternative system characteristics, and anomaly analysis. The final product of the design process includes the sensitivity of service characteristics to system variables, the cost impacts of system variables, and the performance specifications. Table 2 also lists the variables addressed by the APM simulation model that affect cost and performance specifications. After the system designer conducts a series of simulation experiments using the simulation input variables in Table 2, then they can complete the specific geotechnical and foundation analyses, finalize the alignment, conclude station location and joint development agreements and finalize cost targets.

2. LITERATURE REVIEW
Figure 1. APM Planning and Design Process [Dooley, 1980]

2. LITERATURE REVIEW
<table>
<thead>
<tr>
<th>APM Variable</th>
<th>Ridership Variable</th>
<th>Cost Variable</th>
<th>Performance Specification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Station-to-station demand rate by time</td>
<td>In-vehicle and wait times</td>
<td>Capital cost</td>
<td>Meter² per passenger, standee-seated ratio</td>
</tr>
<tr>
<td>Guideway speed limits</td>
<td>In-vehicles</td>
<td>Capital cost</td>
<td>Deboard and board rate</td>
</tr>
<tr>
<td>Station configurations</td>
<td>In-vehicles</td>
<td>Vehicle cost</td>
<td></td>
</tr>
<tr>
<td>Station-to-station distance</td>
<td>In-vehicles</td>
<td>Vehicle cost</td>
<td></td>
</tr>
<tr>
<td>Vehicle speed</td>
<td>Wait time</td>
<td>Vehicle cost, guideway cost</td>
<td></td>
</tr>
<tr>
<td>Vehicle capacity</td>
<td>Wait time</td>
<td>System cost</td>
<td>Time to unload vehicle</td>
</tr>
<tr>
<td>Vehicle loading rates</td>
<td>Transfer time</td>
<td></td>
<td>Maximum delay times</td>
</tr>
<tr>
<td>Vehicle per train</td>
<td>In-vehicle and wait times</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Minimum safe headway</td>
<td>In-vehicle and wait times</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Route headway</td>
<td>Wait time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transfer points</td>
<td>Wait time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Minimum and maximum door open times</td>
<td>Wait time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Failure conditions</td>
<td>Transfer time</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Output</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wait-time distribution</td>
<td>Wait time reliability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>In-vehicle time distribution</td>
<td>In-vehicle time reliability</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maximum station queue</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Load factors</td>
<td>Fare</td>
<td>Station cost</td>
<td>Maximum wait</td>
</tr>
<tr>
<td>Vehicle kilometers</td>
<td>Fare</td>
<td></td>
<td>Platform size</td>
</tr>
<tr>
<td>passengers served</td>
<td>Fare</td>
<td>Operation and maintenance cost</td>
<td></td>
</tr>
<tr>
<td>peak vehicles</td>
<td>Fare</td>
<td>Capital cost, operation and maintenance cost</td>
<td></td>
</tr>
</tbody>
</table>
2.3 Requirements Analysis and Concepts Development

The following discussion briefly reviews some concepts development used in this research:

(1) Level of Service Analysis
(2) APM Demand Analysis
(3) Capacity Analysis
(4) Flow Analysis
(5) Energy Consumption Analysis

2.3.1 Level of Service Analysis

Level of service (LOS) is a general term that includes both the characteristics of service offered and the dependability of that service. In the literature, the LOS of transit systems has been measured and reported with different measuring devices based on the nature of available data and the nature of the analysis. The basic concept of LOS for transit has been discussed in the Chapter 12 of Highway Capacity Manual [HCM, 1994]. The determinants of LOS could be waiting time, travel time, vehicle occupancy, service frequency, level of convenience and comfort, cost, safety, speed and so on. However, for the purpose of this research the measures of APM system LOS are made of the following factors:

(1) a quantitative measures of journey time including waiting, processing, in-vehicle and out-of-vehicle time, and passenger space allocations in the vehicles, at waiting areas or at walking areas that are shown on Table 3; and

(2) system reliability, maintainability and availability to ensure on-time arrival at the airport or terminal and reduced risk of missing a flight.

Reliability performance estimates are made during the design and planning process for use in establishing contract service reliability and availability requirements. Reliability

2. LITERATURE REVIEW
<table>
<thead>
<tr>
<th>Peak-Hour LOS</th>
<th>Approx. m² / Pax</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1.43 or more</td>
</tr>
<tr>
<td>B</td>
<td>0.93</td>
</tr>
<tr>
<td>C</td>
<td>0.70</td>
</tr>
<tr>
<td>D</td>
<td>0.46</td>
</tr>
<tr>
<td>E-1</td>
<td>0.37</td>
</tr>
<tr>
<td>E-2 (Max. scheduled load)</td>
<td>0.31</td>
</tr>
<tr>
<td>F (crush load)</td>
<td>under 0.31</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LOS</th>
<th>Approx. m² / Pax</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2.70 or more</td>
</tr>
<tr>
<td>B</td>
<td>2.30</td>
</tr>
<tr>
<td>C</td>
<td>1.90</td>
</tr>
<tr>
<td>D</td>
<td>1.50</td>
</tr>
<tr>
<td>E</td>
<td>under 1.50</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LOS</th>
<th>Approx. m² / Pax</th>
<th>Approx. m² / Pax</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>3.72 or more</td>
<td>1.89 or more</td>
</tr>
<tr>
<td>B</td>
<td>2.23</td>
<td>1.39</td>
</tr>
<tr>
<td>C</td>
<td>1.49</td>
<td>0.93</td>
</tr>
<tr>
<td>D</td>
<td>1.02</td>
<td>0.65</td>
</tr>
<tr>
<td>E</td>
<td>0.56</td>
<td>0.37</td>
</tr>
<tr>
<td>F</td>
<td>under 0.56</td>
<td>under 0.37</td>
</tr>
</tbody>
</table>
has been expressed as the probability that a system, subsystem or component/assembly will perform its designated function within specified tolerances, without catastrophic failure, under specified conditions over a given interval time. In general, an APM system is composed of the following functional subsystems:

(1) Structures and track work
(2) Stations and APM equipment
(3) Vehicles
(4) Automatic train control
(5) Vehicle train control
(6) Power distribution
(7) Supervisory control and data system
(8) Communication
(9) Maintenance and storage

Each subsystem of an APM system is also comprised of components, subassemblies and functional assemblies, all of which are subject to failure which may result in the blockage, or impending blockage, of operations or reduced capacity operations.

2.3.2 APM Demand Analysis

In order to size the APM link and its stations, and to estimate equipment needs, the demand characteristics need to be considered first. In the APM planning and design process, defining APM systems which satisfy baseline requires demand profiles as an input in the APM simulation model. Basically, the solution requires the distribution over a period of time of the demand. A convenient time unit is a 24 hour period. It is certain that demand in these systems, like any other transportation system, will not be uniform over the day. Random arrivals of passengers to the APM station could result in peak and off-peak demands. Obviously, the peak periods will govern the design of these systems, with
respect to such important system parameters as vehicle capacities, number of vehicles, and station capacities necessary in these systems to meet demand. The off-peak periods will be important for maintenance scheduling and determining storage facility requirements.

In general, the problem of estimation of demand for APM systems should be incorporated in a airport environment, such as airline schedules, sizes of terminal facilities, and number of stations passed. The best indication for general peaking effects in these systems at airports is the airplane arrival and departure schedule data. Passenger data by hour is obtained from each airline. However, this type of data is not even the same for everyday schedules. Moreover, there are other potential ridership resources including follows:

1. The number of individual transferring passengers move directly from the arrival gate to the departure gate without baggage claims, and ticketing.
2. The percentage of visitors, crews, and workers is considered to be serviced.
3. The passengers travel between terminals and remote facilities, such as parking areas, rental car areas, city transit stations, hotels, and so on.

2.3.3 Capacity Analysis

Generally Speaking, APM system capacity is defined as the maximum number of passengers that can be transported on systems past a fixed point in one direction per unit of time (usually 1 hour). An APM system may be defined as a system primarily consisting of two components: guideway and station. Therefore, this capacity is determined by guideway and station capacities. The smaller of the two represents APM system capacity. In this section, the APM system capacity analysis involves these two types of capacities:

1. Guideway capacity
2. Station capacity

2. LITERATURE REVIEW
2.3.3.1 Guideway Capacity

Guideway Capacity is a term used to denote the maximum number of passengers that can be transported in vehicles physically past a fixed point during a given period of time under a given conditions. For most AFM systems, the guideway capacity is usually constrained by a number of factors, such as vehicle performance, the availability of sufficient vehicles to maintain a desired capacity over a sustained period, acceptable levels of service, and safety considerations.

Guideway capacity issues generally involve determining either the maximum number of passengers that can be transported on a facility during a given period of time, the number of vehicles required to provide a stated throughput for a given period of time, or the maximum number of passengers that can be achieved with a given fleet size. These three issues can be investigated using some fairly elementary relationship.

First, possible guideway capacity ($C_w$) is defined by:

$$C_w = f \cdot n \cdot C_v = \frac{3600nC_v}{h_w} \quad (2.1)$$

where

$C_w =$ guideway capacity in passengers/hour

$C_v =$ vehicle capacity in passengers/vehicle

$h_w =$ headway (time separation) between successive transit units in seconds

$f =$ frequency of service in transit unit per hour

$n =$ the number of vehicles per transit unit ($n = 1$ for single unit operation)

Thus, the maximum number of vehicles per transit unit (TU), the passenger capacity of individual vehicles, and the minimum possible headway are the important parameters that determine possible guideway capacity. The headway control strategy is a key factor to influence guideway capacity in the design of APM systems. Headway varies with several

2. LITERATURE REVIEW
operational factors such as primarily speed, vehicle braking characteristics, and level of service provided. Based on safe stopping distance criteria as the minimum distance between transit units, the minimum headway \((h_{w_{\text{min}}})\) and guideway capacity \((C_w)\) are given by:

\[
h_{w_{\text{min}}} = \frac{nl + s_0}{v} + t_r + \frac{v(b_1 - b_2)}{2b_1b_2} \tag{2.2}
\]

\[
C_w = \frac{3600nC_r}{v} + t_r + \frac{v(b_1 - b_2)}{2b_1b_2} \tag{2.3}
\]

where

- \(l\) = vehicle length in m/vehicle
- \(s_0\) = distance between stopped transit units in m
- \(t_r\) = reaction time in seconds
- \(b_1\) = leading vehicle (LTU) breaking rate in m/s²
- \(b_2\) = following vehicle (FTU) breaking rate in m/s²
- \(v\) = velocity in m/s

The values of \(b_1\) and \(b_2\) depend upon the safety regimes of transit vehicle movement, which represent different degrees of operating safety [Vuchic, 1981]. Table 4 shows different safety regimes of transit operation. For each operating regime the minimum headway is obtained by inserting the appropriate values of \(b_1\) and \(b_2\) into Eq. (2.2); then the guideway capacity \((C_w)\) is derived by Eq. (2.3).

To determine the vehicle fleet size required to provide a specified guideway capacity, the following relationships can be utilized. First, the cycle time, or round trip time, depends upon the total distance to be traveled and the average speed:

\[
c = \frac{2L}{V_a} \text{ seconds} \tag{2.4}
\]

2. LITERATURE REVIEW
where

\[ c = \text{cycle time (the time required for a transit unit to complete one round trip including station dwell time) in seconds} \]

\[ L = \text{one-way route length in m} \]

\[ V_a = \text{average vehicular speed over the entire route in m/s} \]

Neglecting jerk and assuming constant acceleration and deceleration, the average vehicular speed (\( V_a \)) between stations is given by:

\[
V_a = \frac{S}{\frac{S}{V} + \frac{V}{2a} + \frac{V}{2d} + T} \tag{2.5}
\]

where

\[ V_a = \text{average vehicular speed in m/s} \]

\[ S = \text{station spacing in m} \]

\[ a = \text{acceleration in m/s}^2 \]

\[ d = \text{deceleration in m/s}^2 \]

\[ V = \text{operating speed or cruise speed in m/s} \]

\[ T = \text{station dwell time in seconds} \]

If station spacing is not uniform throughout the line, the equation for average vehicular speed in one direction over the entire route can be developed by:

\[
V_a = \frac{L}{\frac{L}{V} + k \left( \frac{V}{2a} + \frac{V}{2d} \right) + \sum_{i=1}^{k} T_i + r} \tag{2.6}
\]

where

\[ k = \text{number of stations} \]

2. LITERATURE REVIEW
Table 4. Safety Regimes of Transit Operation: Combinations of Deceleration Types for Successive Transit Units [Vuchic, 1981]

<table>
<thead>
<tr>
<th>Regime Designation</th>
<th>b₁ (LTU)</th>
<th>b₂ (FTU)</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Way a Station a</td>
<td>∞</td>
<td>bₙ</td>
<td>Absolute safety and comfort; possible overdesign</td>
</tr>
<tr>
<td>b Station b</td>
<td>bₑ</td>
<td>bₙ</td>
<td>Safety high, but not absolute for all values of be and bn</td>
</tr>
<tr>
<td>c Station c</td>
<td>∞</td>
<td>bₑ</td>
<td>Safety acceptable for most, but not for automated systems; For bₑ &lt; 2bₙ, safer than regime b, discomfort for FTU passengers</td>
</tr>
<tr>
<td>d</td>
<td>∞</td>
<td>∞</td>
<td>In emergencies safety inadequate</td>
</tr>
<tr>
<td>d₁ Station bₑ</td>
<td>bₑ</td>
<td>bₑ</td>
<td></td>
</tr>
<tr>
<td>d₂ Station bₙ</td>
<td>bₙ</td>
<td>bₙ</td>
<td></td>
</tr>
<tr>
<td>e Station no</td>
<td>no</td>
<td>no</td>
<td>Hypothetical: continuous train</td>
</tr>
<tr>
<td></td>
<td>breaking</td>
<td>breaking</td>
<td></td>
</tr>
</tbody>
</table>

Notes:
- b = bₙ : Normal braking, which is applied in regular operation. The maximum value of bₙ is limited by the safety and comfort of standing passengers.
- b = bₑ : Emergency braking, for which the maximum braking capability of vehicle is applied (often utilizing a special brake, such as a magnetic brake). Passengers experience discomfort and some safety hazard.
- b = ∞ : Instant stop ("brick wall") represents a catastrophic collision. It is only analyzed for providing safe stopping for the FTU when the LTU has such as accident
\[ r = \text{terminal time to turnaround in seconds} \]
\[ L = \text{route length in m} \]

During the time required for one transit unit to complete one round trip, the number of transit units required in the system is given by:

\[ u = \frac{c}{h_w} \tag{2.7} \]

where

\[ u = \text{the number of transit units in the system (single or multiple unit)} \]

Thus, if the headway is shorter, the number of transit units or vehicles required to maintain service frequency will be greater. Therefore, total fleet size required in the system is given by:

\[ N = \left(\frac{c}{h_w}\right)n = \frac{2nL}{h_w V_a} \tag{2.8} \]

where

\[ N = \text{total fleet size required (total vehicles required)} \]

In other words, the number of vehicles required to provide a specified guideway capacity varies inversely with headway and average speed. Finally, if the fleet size is limited, the total number of transit units available, possible headway and capacity are calculated by:

\[ u' = \frac{N'}{n} \tag{2.9} \]

\[ h = \frac{c}{u'} = \frac{cn}{N'} \tag{2.10} \]

2. LITERATURE REVIEW
\[ C'_w = \frac{3600nC_v}{h} = \frac{3600C_vN'}{c} = \frac{1800C_vV_aN'}{L} \]  

(2.11)

where

- \( u' \) = the total number of transit units available
- \( N' \) = the number of vehicles available
- \( C'_w \) = possible capacity for a limited fleet size in passengers/hour

Thus, three basic relationships derived above are shown as follows [Canadian Transit Handbook, 1985]:

1. Maximum passengers per hour without vehicle restrictions:

\[ C_w = f_n C_v = \frac{3600nC_v}{h_w} \]  

(2.12)

2. Fleet size required to provide a specified throughput:

\[ N = \frac{2nL}{h_wV_a} \]  

(2.13)

3. Maximum passengers per hour for a limited fleet size:

\[ C'_w = \frac{1800C_vV_aN'}{L} \]  

(2.14)

2.3.3.2 Station Capacity

Station capacity is defined as the maximum number of passengers transported in vehicles stopping at a station per direction during a given period of time. Station capacity governs the capacity of a transit line in a vast majority of cases due to the following relationships [Vuchic, 1981]:

1. Along the system line, the longest station headway \( (h_{a_{\text{max}}}) \) between successive transit units at stations determines the system capacity.

2. LITERATURE REVIEW
2. Station headway is greatly influenced by standing time $t_s$ (or dwell time).

3. If stations are on-line (that is, no passing is permitted and all vehicles stop at all stations), station capacity may influence system capacity. In the case of off-line stations, where through vehicles (such as express or limited stop vehicles) are allowed to pass stopped vehicles, capacity will be higher.

4. Assuming similar design and operating conditions at all stations, the “busiest station”, which handling the highest passenger volume, determines the system capacity.

Station headways between successive transit units consist of two groups of elements:

1. time intervals of vehicle motion (i.e., acceleration of leading transit unit and deceleration of following transit unit); these depend on vehicle dynamic characteristics, operating regime and safety requirements.

2. station dwell time, a factor in which there can be considerable variation depending on:
   - the number of passengers boarding and alighting
   - fare-collection method (Fare collection may not be available in airport terminal.)
   - the number of loading bays
   - door arrangements, design and number
   - seating arrangements within the vehicle

Based on the operating safety regimes shown on Table 4, the minimum headway ($h_{s\min}$) and station capacity ($C_s$) are given by:

$$h_{s\min} = t_s + t_r + \frac{2nlb_1}{\sqrt{a(a+b_1)}} + \frac{nl}{v} + \frac{v}{b_2}$$  \hspace{1cm} (2.15)

2. LITERATURE REVIEW
\[ C_s = \frac{3600nC_v}{t_s + t_r + \sqrt{\frac{2nlb_1}{a(a + b_1)} + \frac{nI_v}{v} + \frac{v}{b_2}}} \]  

(2.16)

where

\[ t_s = \text{station dwell time in seconds} \]

\[ a = \text{vehicle acceleration rate in m/s}^2 \]

As shown in the preceding sections, for a particular transit technology, the number of vehicles per transit unit, average vehicular speed, individual vehicular capacity, and minimum possible headway are basic parameters that affect design capacities and fleet requirements. These parameters in turn are affected by other factors such as the minimum headway for guideway capacity depends upon operating speed, acceleration and deceleration rates, vehicle characteristics and so on. Thus, the factors that ultimately have direct or indirect influences on design capacities can be grouped on Table 5 [Canadian, 1985].

2.3.4 Flow Analysis

A number of different methods widely used to analyze the flows are as follows:

1. Application of Queuing Theory
2. Fluid Approximation
3. Modeling and Simulation

Queuing theory can be employed to analyze and solve various problems related to storage length, waiting time and number of servers at a system. All stations may be modeled as service facilities using queuing models. Queuing theory is used to estimate the queue size and delay on the platform. The passenger or vehicle arrival gaps at a station follow different probability distributions, such as uniform, poisson, negative binomial or

2. LITERATURE REVIEW
Table 5. Factors to Influence Transit Capacity [Canadian Transit Handbook, 1985]

1. **Vehicle Characteristics**
   - size of vehicle fleet (i.e., number of available vehicles)
   - allowable number of vehicles per transit unit (i.e., single unit or multiple unit)
   - vehicle dimensions
   - seating configuration
   - number and location of doors
   - acceleration and deceleration
   - maximum speed

2. **Right-of-way Characteristics**
   - cross section design
   - degree of separation from other traffic
   - intersection design (at grade or grade separated)
   - horizontal and vertical alignment

3. **Stop Characteristics**
   - spacing of stops
   - on-line or off-line
   - method of fare collection
   - high level or low level loading
   - length of loading bays or platforms
   - turnaround facilities at terminals

4. **Traffic Characteristics**
   - volume and nature of other traffic (on-share right-of-way)
   - cross traffic at intersections (if at grade)

5. **Method of Headway control**
   - type of control
   - separation standards for safety

2. **LITERATURE REVIEW**
exponential, depending upon the passenger or vehicle arrival characteristics at the station for a given period. Since the passenger behavior is highly dynamic in the station environment, the queuing theory which is difficult to apply for steady state conditions and may not be very precise using the mathematical probability equations. However, queuing theory still can provide useful and reasonable estimates of queue size and delay using simple formulations such as follows:

(1) For a single-sever queuing system (s=1) with poisson arrival distribution, negative exponential service times, and constant arrival and service rate, the steady state measures of effectiveness are:

\[ L = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu - \lambda} \]  
\[ (2.17) \]

\[ L_q = \frac{\lambda^2}{\mu(\mu - \lambda)} \]  
\[ (2.18) \]

\[ W = \frac{L}{\lambda} \]  
\[ (2.19) \]

\[ W_q = \frac{L_q}{\lambda} \]  
\[ (2.20) \]

where

\[ P_n = \rho^n P_0 = \left(\frac{\lambda}{\mu}\right)^n P_0 \]  
\[ (2.21) \]

\[ \rho = \frac{\lambda}{\mu} \]  
\[ (2.22) \]

\[ P_n = \text{probability that } n \text{ calling units in the queuing system} \]
\[ P_0 = \text{probability that no one in the queuing system} \]
\[ L = \text{average number of calling units in the queuing system} \]
\[ L_q = \text{average queue length in the queue} \]
W = average waiting time for calling units in the queuing system

Wt = average waiting time for calling units in the queue

λ = arrival rate of calling units

μ = service rate for a server

ρ = utilization factor, which is less than 1

(2) For a multi-sever queuing system (s>1) with poisson arrival distribution, negative exponential service times, and constant arrival and service rate, the steady state measures of effectiveness are:

\[ L_q = \frac{\rho P_0 (ps)^s}{s!(1-\rho)^2} \]  \hspace{1cm} (2.23)

\[ L = L_q + \frac{\lambda}{\mu} \]  \hspace{1cm} (2.24)

\[ W_q = \frac{L_q}{\lambda} \]  \hspace{1cm} (2.25)

\[ W = \frac{L}{\lambda} \]  \hspace{1cm} (2.26)

where

\[ P_n = \frac{(ps)^s}{n!} P_0 \hspace{1cm} \text{if } 0 \leq n \leq s \]  \hspace{1cm} (2.27)

\[ P_n = \frac{\rho^s s^s}{s!} P_0 \hspace{1cm} \text{if } n \geq s \]  \hspace{1cm} (2.28)

\[ P_0 = \left[ \sum_{n=0}^{s-1} \frac{(ps)^n}{n!} + \frac{(ps)^s}{s!(1-\rho)} \right]^{-1} \]  \hspace{1cm} (2.29)

s = number of servers (s > 1)

n = state of the queuing system (no. of calling units in the queuing system)

2. LITERATURE REVIEW
Unlike the queuing theory using complex probability and mathematical equations, the fluid flow approximation is another method for the analysis of queuing systems that treats servers as reservoirs and users as water flowing from reservoir to reservoir [Setti, 1994]. This approximation gives quite reasonable results for rather general queuing systems. It is assumed that the discrete and random arrivals can be approximated by a non-random continuum as if it were a fluid flowing into a reservoir, and the departure process is approximated in a similar way like fluid flowing out of a reservoir. Users arriving at the system are represented by an equivalent fluid into the first reservoir. The volume of liquid accumulated in the reservoir represents the queue and the fluid discharged by the first reservoir into the next one corresponds to users leaving that service channel.

In a system of tandem or series queues, let \( i_k(n) \), a function that describes the input flow into the \( k \)th reservoir, represent arrivals at the entrance of the \( k \)th server during consecutive time intervals of duration \( \Delta t \). Also let the departures from the \( k \)th server during consecutive \( t \) time intervals be represented by \( o_k(n) \), a function that represents the output flow from the \( k \)th reservoir. Therefore, the volume of water contained in the reservoir or the queue length may be expressed by:

\[
q_k(n) = \sum_{i=0}^{n} [i_k(t) - o_k(t)] \Delta t \quad (2.30)
\]

where

- \( q_k(n) \) = queue length at the \( k \)th server at the \( n \)th time interval
- \( i_k(n) \) = input flow to \( k \)th server during interval \( t \)
- \( o_k(n) \) = output flow of \( k \)th server during interval \( t \)

The flow continuity condition within servers means that the variation in the queue equals to the difference between input and output flows:

2. LITERATURE REVIEW
\[ q_k(n) - q_k(n - 1) = i_k(n) - o_k(n) \]  \hspace{1cm} (2.31)

The condition of continuity of flow between successive servers means the input flow into kth reservoir during the nth time interval is the output flow from the jth reservoir during the previous time interval:

\[ i_k(n) = o_j(n - 1) \]  \hspace{1cm} (2.32)

If jth reservoir is connected to more than one reservoir, Eq. 2.32 may be expended to express the condition of continuity:

\[ i_k(n) = x_k o_j(n - 1) \]  \hspace{1cm} (2.33)

\[ i_l(n) = x_l o_j(n - 1) \]  \hspace{1cm} (2.34)

\[ i_m(n) = x_m o_j(n - 1) \]  \hspace{1cm} (2.35)

provided that \( x_k + x_l + x_m + \ldots = 1 \). The condition of continuity of flow between successive servers expressed by Eq. 2.32 guarantees that the total flow entering the system will leave it.

Many transportation systems are so complex that it is difficult for the human mind to follow the complex interactions which occur. Modeling and simulation of such systems on a digital computer provides the means for engineering and management personnel to help them plan, design, and operate actual systems. They can be used test the effects of changes in the flow characteristics, service facilities, and system operating procedures without interfering with the current operations of the real system. A simulation models is built based on a programming language. Therefore, the choice of the a proper programming language is quite important.

2. LITERATURE REVIEW
2.3.5 Description of Selected Simulation Approaches

The System Operations Studies (SOS) Models are selected for APM simulation studies [Dooley, 1979]. The models are used to simulate general purpose automated transit systems. Therefore, the models may be applicable to simulate the airport APM systems within certain limits. However, some approaches proposed in the models may contribute to fulfill this research.

The SOS models are computer-based analysis tools developed in 1978 by the Transportation Systems Center of U. S. Department of Transportation and its contractor, General Motors Transportation Systems Division and IBM Federal Systems Division. The SOS models are special demand and supply analysis tools for the automated guideway transit (AGT) system. The research and development of these models were sponsored by the Urban Mass Transportation Administration (UMTA). The SOS models have been designed to analyze the complete spectrum of AGT system applications and problem areas. The SOS models include three analytical models: the feeder system model (FSM), the system cost model (SCM) and the system availability model (SAM), and four simulation models: the system planning model (SPM), discrete event simulation model (DESM), detailed station model (DSM) and detailed operational control model (DOCM). All models are written in structured FORTRAN programming language.

The SOS models can analyze alternative AGT system designs for a given scenario. It is assumed that the general locations of stations, a zone-to-zone demand, and certain characteristics of AGT and feeder systems are known before modeling. Given this information, the input processor of the DESM can generate the travel time impedances expected or desired between stations in the AGT network. The overview of all models is described as follows:

The feeder system model (FSM) is used to map the serviceable zone-to-zone demand onto the stations of AGT network and to calculate utilization and performance of

2. LITERATURE REVIEW
the feeder transit service. In the FSM process, a series of station-to-station demand matrices will be generated for the SPM and DESM. To determine the demand at each station, the FSM finds the station pair which provides the best zone-to-zone travel time for a particular type of feeder service and the nominal AGT station-to-station travel times.

The system planning model (SPM) is a model which establishes the relationship between specific system parameters and measures of system effectiveness. The model uses average station-to-station demand from the FSM and network configuration data to estimate the static and dynamic vehicle and passenger flows on network links trip times, and vehicle efficiencies for a range of vehicle types. The SPM models the passenger and vehicle flows using the fluid approximation for queues.

The detailed station model (DSM) is an event simulation model which provides a means of sizing the transit station based on both vehicle and passenger handling capacities and alternative station configurations. The model uses a discrete event modeling approach to represent the interrelated queuing processes associated with vehicle and passenger activities in a single station. The model is used to evaluate queue sizes and transit times for station processes.

The detailed operational control model (DOCM) is used to simulate the detailed motion of automated vehicles moving on a guideway link, traveling through an intersection, and/or performing a merge function. The DOCM typically generates and stores the following performance measures: vehicle travel time, the probability of violating headway, acceleration, and jerk limits, queue lengths, propulsive and braking work done during the simulation period, and measures of merge and failure initiated congestion.

The discrete event simulation model (DESM) is the major SOS analysis tool used to simulate the operations of specified AGT systems over a complete network. The model can represent stochastic arrivals, alternative service policies, path selection algorithms, dispatching strategies, longitudinal and merge control policies, station configurations, 2. LITERATURE REVIEW
empty vehicle management strategies, and failure conditions. The detailed route, link, station, trip statistics, and a time log for all passenger trips will be generated in the DESM.

The system availability model (SAM) is designed to calculate vehicle and passenger availability measures and required maintenance and standby fleet size using system operating data, network descriptions, subsystem failure rates, demand data from the SOS database, and trip log data from the DESM.

The system cost model (SCM) is used to calculate the cash flow process for financing and operating an AGT system. The AGT and feeder utilization measures generated by the DESM and FSM are used together with network and system characteristics data by SCM to develop estimates of the life cycle cost (capital, operating, and maintenance) and environmental impacts (energy consumption, pollution, and land use).

In summary, the SOS models can be grouped on three levels for AGT system analyses under development. There are the high-level coarse models (FSM, SPM, SAM, and SCM), the detailed subsystem models (DSM, DOCM), and the detailed system model (DESM):

1. The detailed models of station (DSM) and link/merge/intersection control (DOCM) are used either to provide data for modeling at the DESM level or for preliminary analysis of critical components before modeling at the system level.

2. A discrete event simulation model (DESM) is the major SOS tool in the analysis of subsystem and algorithm alternatives in a system context and is also the basic tool for the system level evaluation of the more complex and sophisticated system management strategies.

3. A set of high-level coarse models are used to analyze the performance of the simpler system (SPM), to estimate the system availability (SAM), to calculate system cost (SCM), and to estimate feeder service characteristics (FSM).

2. LITERATURE REVIEW
2.3.6 Energy Consumption Analysis

The power required to move the vehicle at speed \( V \) is calculated by:

\[
P = \frac{T(V)}{\eta}
\]  \hspace{1cm} (2.36)

where

\( P \) = power developed by the engine(s) in watts
\( T \) = the tractive force required to overcome the resistance forces in newtons (N)
\( V \) = speed in m/s
\( \eta \) = an efficient factor

The power is the time rate of doing work which has units of energy consumption. Therefore, the energy consumed can be calculated by:

\[
E = 3.6 \times 10^6 \int_0^t P\,dt
\]  \hspace{1cm} (2.37)

where

\( E \) = energy consumed in kilowatt-hours (kWh)
\( P \) = power in watts
\( t \) = the operating time in seconds

The major contributors of total energy consumption per APM transit unit between two stations are: (1) aerodynamic drag, (2) rolling resistance, (3) guideway resistance (flanges, joints), and (4) alignment resistance (gradient, curvature). For a transit unit traveling in a straight and level track, the resistance is usually estimated using the Davis’ equation:

\[
R_{(a+r)} = K_0 + \frac{K_1}{w} + B(V) + \frac{CAV^2}{wn}
\]  \hspace{1cm} (2.38)
where

\[ R_{(a+t)} = \text{inherent resistance (aerodynamic + rolling resistances) in lb/ton} \]

\[ w = \text{average load per axle in tons} \]

\[ A = \text{cross sectional area in ft}^2 \]

\[ B = \text{an experimental coefficient due to guideway conditions} \]

\[ C = \text{drag coefficient or shape factor} \]

\[ V = \text{velocity of vehicle in miles/hour (mph)} \]

\[ n = \text{number of axles} \]

\[ K_0 \text{ and } K_1 \text{ are constant coefficients with magnitudes 1.3 and 29, respectively. If} \]

Davis’ equation is expressed in SI units where inherent resistance \((R_{(a+t)})\) is in Newtons/Newton (N/N), average load per axle \((w)\) is in Newtons, cross sectional area \((A)\) is in m\(^2\), and velocity \((V)\) is in m/s, the following relationship applies:

\[ R_{(a+t)} = 5 \times 10^{-4} K_0 + 4.4480 \frac{K_1}{w} + 1.1187 \times 10^{-3} B(V) + 239.6904 \frac{CAV^2}{wn} \quad (2.39) \]

This equation does not include the resistance caused by alignment. B and C are coefficients applicable to different types of equipment. Table 6 lists values of A, B, and C that are reasonable for current transit equipment. In addition, the gradient resistance assumed in going up a hill is computed by:

\[ R_{(g)} = 9.81mg \left( \frac{i}{100} \right) \quad (2.40) \]

where

\[ R_{(g)} = \text{gradient resistance in Newtons} \]

\[ mg = \text{gross vehicle weight in kgf} \]

\[ i = \text{the instantaneous gradient in percent} \]

2. LITERATURE REVIEW
### Table 6. Values for Use in the Davis' Equation

<table>
<thead>
<tr>
<th>Equipment</th>
<th>A (m²)</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>Leading car or</td>
<td>8.36 - 11.15</td>
<td>0.03</td>
<td>0.00160</td>
</tr>
<tr>
<td>locomotive</td>
<td>(90 - 120 ft²)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Trailing cars</td>
<td>8.36 - 11.15</td>
<td>0.03</td>
<td>0.00034</td>
</tr>
<tr>
<td></td>
<td>(90 - 120 ft²)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. LITERATURE REVIEW
3. METHODOLOGY

3.1 Introduction

This chapter briefly describes the methodology used for conducting this research. The main objective of the methodology is to develop a reasonably realistic, easily implemented, systematically efficient method for planning and evaluation of the airport APM system. Simulation is a method of providing detailed operating information about a real system by imitating its operation. The computer model used in this research is a discrete-event simulation model which represents the movement of individual passengers and vehicles in the APM system. This chapter describes a basic definition of simulation, the development of the simulation model, the approach in a simulation study, and the use of Extend as a simulation tool.

3.2 Systems, Models and Simulation

A system is defined to be a collection of entities that act and interact together toward the accomplishment of some logic end [Schmidt, 1970]. A system may contain a set of related subsystems to form a unit. Therefore, in this research the airport APM system is simply defined to be that portion of the airport consisting of the transit units (TUs) and the passengers as basic entities. These entities are characterized by attributes that may themselves be related. For example, each TU has attributes such as the number of vehicles and capacity of a vehicle. Each passenger has attributes such as travel time and destination station. The capacity of the TU is related to the number of vehicles, which in turn may be related to the passenger’s travel time.

The first step in studying a system is to build a model. Concisely, a model can be thought of as a representation of the system of interest. It is used to described how a real-world activity will perform, to investigate possible improvements of the real system, or to
discover the effect of different policies on that system. There are many kinds of models. For example, physical, verbal-descriptive, graphical, mathematical, logical algorithmic and computer models. In many cases, a given system may best be modeled by a combination of these types. A model is always realized to satisfy an objective as the purpose of study. It should be highly faithful to represent the real system in order to accurately reproduce or predict the results of empirical observations. Good model performance also relies on the level of detail and nonambiguity of the model. Sometimes, a total system model can be constructed on the basis of models of subsystems (submodels) to reduce complexity because the submodels are easier to identify. However, submodel integration may involve difficult decisions as to how to treat causality inside a total system model.

Simulation involves designing a mathematical and logical model of a real system and carrying out experiments on it as it progresses through time. For example, the board game Monopoly is a model of a real system. When players are in this game, they are simulating that system. Of the several methods of simulating an APM system, the event based method of simulation is used in this research. Discrete-event simulation concerns the modeling of a system as it evolves over time by a representation in which the state variables change instantaneously at separate points in time [Law, 1991]. From a theoretical standpoint, an APM simulation model should be:

1. goal or purpose directed
2. simple to understand and easy to control and manipulate
3. suitable for a wide range of situations
4. easy to modify or update the model
5. able to evaluate present states and also support evaluation and selection through judgment and predictions
6. effective for process of airport APM service design
7. effective to do analysis of threshold and performance measurement standards

3. **Methodology**
To be useful, the APM simulation should respond as the actual APM system does in its normal operation. The simulation should be used to determine the effect of various changes upon the real system by accurately reproducing the operation of the system under these changed conditions. An important characteristic of the APM simulation is its ability to simulate variations in vehicular speed and station dwell time. Moreover, the simulation should contain random variations which can be turned on and off selectively to give stochastic and deterministic results, respectively depending on the purpose of the model.

3.3 Approach in a Simulation Study

A successful simulation study relies on the methodology used by the modeller and his or her knowledge platform and expertise in the analysis of integrated systems. It is usually better to have a plan before starting modeling. The following six activities or steps are applied in this research to compose a sound simulation study.

(1) Problem Formulation. Successful problem formulation is the most critical step in developing a model of a system. Therefore, a technique to define and model the problem is of great importance. The first issue in problem formulation is to bound the problem, to gather the system information, and to specify the overall objectives to be achieved. After problem has been formulated, conceptual models for the APM system, its functionality, and the required data structures will be developed in this step.

(2) Data Collection and Analysis. Information and data needed by the APM model should be identified and collected. Basically, the major sources of data are APM system records, expert opinion, and field studies. All parameters and variables that affect the measure of APM performance are also identified. For use in the APM simulation model, data are need to estimate values of constants and parameters, provide initial values for all variables, and provide data which simulation outputs can be compared with for validation. Based on valid information and data, operating procedures and probability distributions for

3. METHODOLOGY
the random variables used in the APM simulation model can be specified. All parameters and variables will be discussed in Chapter 4. In this step, the APM simulation model and its subsystem models can be defined.

(3) **Model Development.** The APM simulation model consists of two major subsystem models: APM station model and APM guideway model. Both APM station and APM guideway subsystems have their own subsystems. For example, the platform is one of subsystems in APM station subsystem. To formulate subsystem models in the APM simulation model, a process-oriented approach is used to analyze activities. The process-oriented approach is a method that is often used to define the flow of physical or information entities through the system on the basis of sequences of events (processes) that occur in a pre-defined pattern, and to analyze item processing for system having dynamic properties determined by the flow through the system. The simulation logic as adopted in this research can be broadly divided into two parts: (1) passenger flow logic and (2) vehicular flow logic. The detail of simulation logic will be presented in Chapter 4. After formulating the subsystem models and building their blocks in the Extend environment, the user can combine the subsystem models into a desired APM system model. It is necessary to make pilot runs after models have been completed to verify the functionality of the model. In this step, it is also important to determine the form of outputs to reach a research purpose. A good output format is useful for validating the model.

(4) **Validation and Modification of Models.** The validity of a simulation is a measure of the extent to which the model satisfies its study objectives. Validation involves conducting a series of tests on the input, output, and structure of the model. Common methods for validating are to compare the output of the simulation model to historical data under similar environmental conditions, and to check the reliability of output. Basically, if the output is similar to the historical or field data, the simulation is accepted as a realistic representation of the system. The following output from the APM simulation model is used

3. **METHODOLOGY**
as measures of effectiveness for comparison with the historical data: (1) waiting time
distribution of passengers at APM stations, (2) queue lengths at APM stations, and (3)
volume of passengers in APMs on different routes. However, the procedure may return to
previous steps when the output is not acceptable. The ultimate validation of the APM
simulation model is how well it predicts the operations of a system.

(5) Design Experiments and Running the Simulation. In this step it must be decided
what system designs to simulate. In the APM simulation model, an experimental design
requires observation of the APM system under specific combinations of Extend’s
functional blocks which are capable of representing the system. Then, the APM simulation
model is established using required blocks according to the APM system selected in study.
For a selected APM system design to be simulated, some decisions have to be made on
such issues as initial conditions for the simulation run, the duration time of the simulation
run, and the number of independent simulation runs for sensitivity analysis. The APM
simulation model developed can now be tested and used in actual airport landside
scenarios.

(6) Analysis of Output Data and implementation. Analysis of the simulation results
is to achieve the research objectives. This step brings out some important outcomes such as
the determination of levels of service (LOS) in particular facilities, critical capacity,
maximum queuing, maximum and average waiting times, and so on. The analyst draws
inferences from these outcomes and makes specific operational recommendations on how
the APM system works and how could be improved.

Development of a computer simulation often leads to the discovery that a system
believed to be simple actually is quite complex. However, a simulation study is not a
simple sequential process. At a number of points the process may go back to a previous
step.

3. METHODOLOGY
3.4 Description of Extend

Extend, created by Bob Diamond in 1987, is a powerful and advanced simulation tool for decision support. Extent 3.0.2 is the current version of Extent only for Macintosh computers. This tool is the first simulation application combining powerful features of advanced simulation systems into one package that allows users of any discipline to develop their own libraries of customized simulation tools.

An Extend simulation model consists of a series of interconnected blocks which represent different parts of simulation processes. These blocks can be changed quickly to explore "what-if" questions about the system. In assembling many pre-defined blocks into a single model, the user can use a series of simple block definitions to describe complex processes in a two-dimensional drawing environment. Extend has a flexible iconic user interface, expandable libraries of reusable blocks, hierarchical blocks, animation, a built-in authoring environment, sensitivity analysis, and an excellent interface to make model building fast and easy. The user can add blocks to a model, connect them with the mouse, then run the simulation to achieve graphical or tabular results. Extend users can readily model continuous, discrete event, and mixed-mode system. The important features of Extend are summarized as follows:

1. A full array of building blocks to create model quickly
2. Animation of movies, picture, graphic, or numbers for better presentation
3. A customizable graphical interface showing the relationships in the model
4. Hierarchical modeling to make complex systems easy to build and understand
5. Dialogs and notebooks in blocks allowing to change model values quickly
6. The ability to adjust settings while the model is running
7. Full connectivity with other programs through copy/paste, import/export, publish/subscribe, text files, XCMDs, and so on
8. System optimization using Monte Carlo, batch mode and sensitivity analysis

3. METHODOLOGY
Moreover, Extend’s integrated environment is composed of the following advanced features of the powerful simulation systems:

1. Extend is a multi-application environment where continuous, discrete-event, linear, and non-linear dynamic systems can be modeled.

2. Extend has build-in ModL scripting language much like the C programming language to allow the user build specialized blocks, modify any of Extend’s standard blocks, or create customized animation into a model.

3. All pre-defined blocks can be saved in libraries and easily reused in other models.

4. Extend has over 200 directly accessible functions for integration, statistics, queuing, animation, IEEE math, matrix, sound, arrays, FFT, debugging, XCMOs, string and bit manipulation, I/O, etc.

5. Extend has message sending capabilities that can send messages between blocks interactively for subprocessing.

6. Extend has sophisticated data-passing capabilities to pass values, arrays, or structures composed of arrays of arrays.

7. The model size is limited by the computer system memory size.

8. On-line help is available from help command in the desk accessory menu, from help button in each block’s dialog, or from the Help block in a model.

Extend libraries hold pre-defined blocks. The purpose of a library is to be an easy-to-use repository for the definitions of the blocks that are used in simulation models. Extend 3.0.2 has four standard libraries labeled Generic, Discrete-Event, Plotter, and Animation libraries. Some blocks of these standard libraries are used in this research and listed in Appendices A, B, and C.
The General library has blocks that perform basic functions as math, decision handling and input/output. It can be used to solve almost any continuous modeling problems. Some blocks of the Generic library also can be used in discrete-event models.

The Discrete-Event library contains all the basic tools for building models that use queues, servers, item-specific attributes, and priorities. The Discrete-Event blocks are useful for simulating queuing theory, computer and communications networks, service industry waiting lines, and more. These blocks can only be used in event-driven (non-continuous) models because they tend to change the timing of the simulation model.

The Plotter library holds blocks for plotting the results in any type of simulation model. Some blocks are specific to continuous or discrete-event models while others can be used with either. These blocks can show scatter and line plots, moving strip charts, histogram charts, and so on.

The Animation library holds blocks that are used for animating hierarchical blocks. There are only two blocks in this library: Animate Item and Animate Value blocks. These blocks serve as important debugging tools during the simulation and are useful in visual verification of the model.

3. METHODOLOGY
4. MODEL DESCRIPTION

4.1 Introduction

This chapter begins with the description of model specifications. The description defines the APM system in terms of its important components and their interrelationships. The simulation model logic is then presented and is illustrated by flow diagrams in which each block corresponds to an event or a process. The function of each block is also described at the end of this chapter.

4.2 Model Specifications

The model specifications for the APM system to be modeled contain definitions in terms of entities (system components), activities, attributes, events and their relationships. The concepts of event, process, and activity are important when constructing a model of a system. An event signifies a change in state of an entity. A process is a sequence of events ordered on time. An activity is a collection of operations that transform the state of an entity [Fishman, 1973]. So an event is the occurrence of an activity at a particular instant of time.

4.2.1 System Components

The model only considers entities (system components) of the real system environment which could cause significant effects in the decision making process. Each type of entity naturally has a distinct set of attributes associated with it. Generally, an APM system contains several major entities falling in two categories: (1) station model and (2) guideway model. The station model is defined in terms of seven entities: (1) circulation area/passageway, (2) elevator, (3) entrance/exit, (4) escalator, (5) platform, (6) sidewalk, and (7) stairway. The guideway model is defined in terms of five entities: (1) 2-way switch facility, (2) merge/diverge guideway, (3) pinched loop guideway, (4) single-lane...
guideway, and (5) turnaround facility. Another two essential components of the APM model are passenger and TU flows processed through the whole system. These entities and their attributes associated with the system are listed in Table 7. In the APM simulation model, most entities are built as an individual or hierarchical blocks in which an event, an activity or a process occur. The required data standing for their attributes are input directly in their block structure windows.

4.2.2 System Activities and Associated Events

The possible activities of the system that are currently implemented in the APM simulation model are described in Table 8. It should be realized that the activities of passengers arriving at the original stations and leaving destination stations in the system are omitted. These arrival and leave of passengers are treated as scheduled events. Like passenger scheduled events, the arrival activities (mostly at beginning of simulation) of TUs are also treated as scheduled events that generate new TUs into the system. In most cases, the simulation does not delete TUs from the system, that is, the TU departures are not only activities but also events.

Table 9 lists the possible events associated with the system activities. A detailed discussion of events and their associated activities is given in the simulation model logic section.

4.3 Model Assumptions

To create a flexible and reliable simulation model, the following basic assumptions that characterize the APM system were made:

(1) Simulation duration time is specified by the modeller for a particular system.

(2) Simulation requires the passenger origin-destination linkages to be identified and known using Pax-Schedule blocks for a specified time period.

4. MODEL DESCRIPTION
(3) The boarding and alighting passenger flows are mixed and are simulated together with vehicular flow.

(4) The new passengers added in the model are only from stations using Pax-Schedule blocks. The new TUs added in the model can be generated at stations or on the guideways using Veh-Schedule blocks.

(5) Passengers on escalators and sidewalks do not walk on these facilities.

(6) Passengers alight first then board when TUs arrive at stations.

(7) The boarding time per passenger is same to the alighting time as a constant

(8) The simulation model does not consider the transfers.

(9) There are no queue and delay for TUs in the network. The error message will be shown as a queue or delay happens. The safety headway strategies are controlled by vehicle schedule using Veh-Schedule blocks.

(10) The descriptions of the characteristics of all TUs are same. For example, the mass of TUs are all same.

(11) Station dwell time is constant and each TU comes to a complete stop for a full dwell time at every station.

(12) A station can only be located at the end point of a guideway section.

(13) Up to ten TUs can be simulated at one time. If over ten TUs, they can be simulated without creating energy consumption files.

(14) The speed of TUs is not necessary to reach the cruise speed depending on the distance between two stations.

(15) The acceleration rate of a TU is not constant and derived form the combination of tractive effort and resistance.

(16) The braking acceleration rate of a TU is constant.

(17) The total resistance of a TU only considers aerodynamic, rolling, and gradient resistances and is calculated based on the Davis’ equation.

4. MODEL DESCRIPTION
Table 7. Entities and Their Attributes of an APM System

<table>
<thead>
<tr>
<th>Entities</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Passenger</td>
<td>Origin (station number)</td>
</tr>
<tr>
<td></td>
<td>Destination (station number)</td>
</tr>
<tr>
<td></td>
<td>Arrival time</td>
</tr>
<tr>
<td></td>
<td>Travel time</td>
</tr>
<tr>
<td></td>
<td>Boarding</td>
</tr>
<tr>
<td></td>
<td>Alighting</td>
</tr>
<tr>
<td></td>
<td>Average alighting/boarding time (constant)</td>
</tr>
<tr>
<td></td>
<td>TU number (the TU taken by passenger)</td>
</tr>
<tr>
<td>Transit Unit</td>
<td>TU number</td>
</tr>
<tr>
<td></td>
<td>Arrival time</td>
</tr>
<tr>
<td></td>
<td>Departure time</td>
</tr>
<tr>
<td></td>
<td>Direction</td>
</tr>
<tr>
<td></td>
<td>Mass of a TU</td>
</tr>
<tr>
<td></td>
<td>Number of vehicles per TU</td>
</tr>
<tr>
<td></td>
<td>Number of axles per TU</td>
</tr>
<tr>
<td></td>
<td>Average load per axle</td>
</tr>
<tr>
<td></td>
<td>Cross sectional area</td>
</tr>
<tr>
<td></td>
<td>Capacity per vehicle</td>
</tr>
<tr>
<td></td>
<td>Number of effective doors per vehicle</td>
</tr>
<tr>
<td></td>
<td>Door open/close time (constant)</td>
</tr>
<tr>
<td></td>
<td>Area per vehicle</td>
</tr>
<tr>
<td></td>
<td>Station dwell time</td>
</tr>
<tr>
<td></td>
<td>Turnback or not (at a particular station)</td>
</tr>
<tr>
<td>Station</td>
<td>Station number</td>
</tr>
<tr>
<td></td>
<td>Number of guideway lines</td>
</tr>
<tr>
<td></td>
<td>Station dwell time</td>
</tr>
<tr>
<td></td>
<td>Maximum number of TUs at a station</td>
</tr>
<tr>
<td></td>
<td>Number of circulation areas/passageways</td>
</tr>
<tr>
<td></td>
<td>Number of elevators</td>
</tr>
<tr>
<td></td>
<td>Number of entrances/exits</td>
</tr>
</tbody>
</table>

4. MODEL DESCRIPTION
<table>
<thead>
<tr>
<th>Entities</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Station (continued)</td>
<td>Number of escalators</td>
</tr>
<tr>
<td></td>
<td>Number of platforms</td>
</tr>
<tr>
<td></td>
<td>Number of sidewalks</td>
</tr>
<tr>
<td></td>
<td>Number of stairways</td>
</tr>
<tr>
<td>Circulation Area/Passageway</td>
<td>Area of facility</td>
</tr>
<tr>
<td></td>
<td>Maximum number of passengers entering at one time</td>
</tr>
<tr>
<td></td>
<td>Capacity of facility</td>
</tr>
<tr>
<td></td>
<td>Time in walking on facility (random or constant)</td>
</tr>
<tr>
<td>Elevator</td>
<td>Area of facility</td>
</tr>
<tr>
<td></td>
<td>Capacity of facility</td>
</tr>
<tr>
<td></td>
<td>Time to enter/exit facility (random or constant)</td>
</tr>
<tr>
<td></td>
<td>Time in facility (constant)</td>
</tr>
<tr>
<td>Entrance/Exit</td>
<td>Area of facility</td>
</tr>
<tr>
<td></td>
<td>With fare collections/turnstile or not</td>
</tr>
<tr>
<td></td>
<td>Number of fare collections/turnstile</td>
</tr>
<tr>
<td></td>
<td>Delay in fare collections/turnstile</td>
</tr>
<tr>
<td></td>
<td>Average time through entrance/exit (constant)</td>
</tr>
<tr>
<td></td>
<td>Total capacity of doors</td>
</tr>
<tr>
<td>Escalator</td>
<td>Length of facility</td>
</tr>
<tr>
<td></td>
<td>Speed of facility</td>
</tr>
<tr>
<td></td>
<td>Capacity of facility</td>
</tr>
<tr>
<td></td>
<td>Area of facility</td>
</tr>
<tr>
<td></td>
<td>Delay to enter facility (random or constant)</td>
</tr>
<tr>
<td>Platform</td>
<td>Station number</td>
</tr>
<tr>
<td></td>
<td>Capacity of facility</td>
</tr>
<tr>
<td></td>
<td>Area of facility</td>
</tr>
<tr>
<td></td>
<td>Time for alighting/boarding (random or constant)</td>
</tr>
</tbody>
</table>

4. MODEL DESCRIPTION
Table 7. (continued)

<table>
<thead>
<tr>
<th>Entities</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sidewalk</td>
<td>Length of facility</td>
</tr>
<tr>
<td></td>
<td>Speed of facility</td>
</tr>
<tr>
<td></td>
<td>Capacity of facility</td>
</tr>
<tr>
<td></td>
<td>Area of facility</td>
</tr>
<tr>
<td></td>
<td>Delay to enter facility (random or constant)</td>
</tr>
<tr>
<td>Stairway</td>
<td>Capacity of facility</td>
</tr>
<tr>
<td></td>
<td>Maximum number of passengers entering facility at one time</td>
</tr>
<tr>
<td></td>
<td>Area of facility</td>
</tr>
<tr>
<td></td>
<td>Time in walking on facility (random or constant)</td>
</tr>
<tr>
<td>Guideway</td>
<td>Number of 2-way switch facilities</td>
</tr>
<tr>
<td></td>
<td>Number of merge/diverge guideways</td>
</tr>
<tr>
<td></td>
<td>Number of pinched loop guideways</td>
</tr>
<tr>
<td></td>
<td>Number of single-lane guideways</td>
</tr>
<tr>
<td></td>
<td>Number of turnaround facilities</td>
</tr>
<tr>
<td></td>
<td>Number of entrances/exits</td>
</tr>
<tr>
<td></td>
<td>Number of escalators</td>
</tr>
<tr>
<td></td>
<td>Number of platforms</td>
</tr>
<tr>
<td>2-Way Switch Facility</td>
<td>TU number</td>
</tr>
<tr>
<td></td>
<td>Indicator for guideway switching</td>
</tr>
<tr>
<td>Merge/Diverge Guideway</td>
<td>Length of guideway</td>
</tr>
<tr>
<td></td>
<td>Distances to stations from guideway</td>
</tr>
<tr>
<td></td>
<td>Cruise speed (constant)</td>
</tr>
<tr>
<td></td>
<td>Braking acceleration rate (constant)</td>
</tr>
<tr>
<td></td>
<td>Gradient of guideway</td>
</tr>
<tr>
<td></td>
<td>Maximum number of TUs in this guideway at one time</td>
</tr>
<tr>
<td></td>
<td>Waiting on the minor line or not</td>
</tr>
<tr>
<td></td>
<td>Indicator for guideway switching</td>
</tr>
<tr>
<td></td>
<td>Tractive force</td>
</tr>
</tbody>
</table>

4. MODEL DESCRIPTION
<table>
<thead>
<tr>
<th>Entities</th>
<th>Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pinched Loop Guideway</td>
<td>Length of guideway</td>
</tr>
<tr>
<td></td>
<td>Distances to stations from guideway</td>
</tr>
<tr>
<td></td>
<td>Cruise speed (constant)</td>
</tr>
<tr>
<td></td>
<td>Breaking acceleration rate (constant)</td>
</tr>
<tr>
<td></td>
<td>Gradient of guideway</td>
</tr>
<tr>
<td></td>
<td>Maximum number of TUs in this guideway at one time</td>
</tr>
<tr>
<td></td>
<td>Indicator for guideway switching</td>
</tr>
<tr>
<td></td>
<td>Tractive force</td>
</tr>
<tr>
<td>Single-Lane Guideway</td>
<td>Length of guideway</td>
</tr>
<tr>
<td></td>
<td>Distances to stations from guideway</td>
</tr>
<tr>
<td></td>
<td>Cruise speed (constant)</td>
</tr>
<tr>
<td></td>
<td>Breaking acceleration rate (constant)</td>
</tr>
<tr>
<td></td>
<td>Gradient of guideway</td>
</tr>
<tr>
<td></td>
<td>Maximum number of TUs in this guideway at one time</td>
</tr>
<tr>
<td></td>
<td>Tractive force</td>
</tr>
<tr>
<td>Turnaround Facility</td>
<td>Average turnaround time per TU (constant)</td>
</tr>
</tbody>
</table>
| **Station:** | Passenger walks through the entrance  
|             | Passenger walks through exit  
|             | Passenger walks on circulation area  
|             | Passenger walks on passageway  
|             | Passenger walks on stairway  
|             | Passenger takes elevator  
|             | Passenger takes escalator  
|             | Passenger takes sidewalk  
|             | Passenger walks on or stay on platform  
|             | Passenger boards the TU  
|             | Passenger alights from TU  
|             | TU arrives at a station (TUs are not generated at this moment)  
|             | TU departs from a station (in most cases)  
| **Guideway:** | TU is switched to the other guideway  
|             | TU is merged to the other guideway  
|             | TU is diverged to the other guideway  
|             | TU is diverged and then merged to the other guideway (pinched loop)  
|             | TU arrives at a station (TUs are not generated at this moment)  
|             | TU departs from a station (in most cases)  
|             | Reversed TU is turned to the other direction at a station  
|             | Reversed TU is turned to the other direction on the guideway  
|             | TU travels on the guideway  

4. MODEL DESCRIPTION
Table 9. Possible Events Associated with System Activities in the APM Simulation

<table>
<thead>
<tr>
<th>Station</th>
<th>Guideway</th>
</tr>
</thead>
<tbody>
<tr>
<td>New passenger arrives at an</td>
<td>TU reaches switch point</td>
</tr>
<tr>
<td>originating station</td>
<td>TU reaches merging point</td>
</tr>
<tr>
<td>Passenger reaches entrance</td>
<td>TU reaches diverging point</td>
</tr>
<tr>
<td>Passenger reaches exit</td>
<td>TU arrives at the station</td>
</tr>
<tr>
<td>Passenger reaches circulation area</td>
<td>TU departs from the station</td>
</tr>
<tr>
<td>Passenger reaches passageway</td>
<td>Reversed TU begins to turn back at the station</td>
</tr>
<tr>
<td>Passenger reaches stairway</td>
<td>Reversed TU begins to turn back on the guideway</td>
</tr>
<tr>
<td>Passenger reaches elevator</td>
<td>TU departs from beginning of the guideway</td>
</tr>
<tr>
<td>Passenger reaches escalator</td>
<td>TU reaches end of the guideway</td>
</tr>
<tr>
<td>Passenger reaches sidewalk</td>
<td></td>
</tr>
<tr>
<td>Passenger reaches platform</td>
<td></td>
</tr>
<tr>
<td>Beginning of passenger alighting</td>
<td></td>
</tr>
<tr>
<td>Beginning of passenger boarding</td>
<td></td>
</tr>
</tbody>
</table>

4. MODEL DESCRIPTION
4.4 Simulation Model Logic

The APM simulation model, based on a highly graphical interface, reflects the real system environment through a block diagram representing a network. The simulation model logic is based on two flows: (1) passenger and (2) vehicular flows. Thus all the blocks set in the model represent the actual flow pattern of the system environment. The boarding and deplaning passenger flows are mixed as a biflow and are simulated in the model. Using the same passenger biflow paradigm, the inbound and outbound TU flows are simulated together. In addition, the flow patterns of passenger leaving and arriving at each station are simulated simultaneously with the modeling of the TUs’ progress through the guideway network. A process-orientated approach is applied to describe the simulation model logic, in which passengers and TUs are treated as entities that form flows through the system. In developing the model to be used for simulation it is important to determine what characteristics of the system are to be simulated in great detail and others that can be simplified in order to make the model more flexible and easier to manage. The simulation is intended to study LOS of facilities, passenger queuing and delay, and TU speed and energy requirements. Therefore, the operations of each station and the performance of each TU on the guideway are simulated in great detail. The APM simulation model consists of two separate parts or submodels: (1) APM station model and (2) APM guideway model. The simulation model logic is illustrated in these two submodels:

4.4.1 APM Station Model

The identification of the station system is the first step in developing the APM station model. The model suggests that a station is modeled as a series of system component blocks where events occur and each block represents a basic type of functional activity. These functional blocks are listed and described in Section 4.7. The second step is to identify the system components (entities), attributes, activities, and events that occur in

4. MODEL DESCRIPTION
each system component. Within the station system there are two basic entities: (1) passenger and (2) transit unit. The attributes of these two entities and other entities are described in Table 6. The dynamic feature of the model is the activities that occur within each system component block, that is, the activity within the block performed by passengers or transit units. The genesis of an activity is an event which may start or stop an activity. The possible activities and their associated events are listed in Table 7 and 8.

The passengers arrive at their origin station at the time defined in Pax-Schedule blocks and enter a boarding queue through station system components such as the escalator, stairway, and passageway facilities. When the simulated TU arrives, passengers alight if the particular station is their destination and then board if the TU has sufficient space and is headed toward their destination stations. Figure 2 illustrates the passenger flow pattern in a typical station system including system components of one entrance/exit, one passageway, one stairway, one elevator, two escalator, and one shared platform facilities with two guideway lines.

There are three kinds of the Pax-Schedule blocks that generate arrival of passengers. Passengers arrival independently of each other based on arrival rates, specific times, or distributed interarrival times which can be specified in the different Pax-Schedule blocks. If necessary these three blocks also can be combined or reused for the model purpose. After the arrival times of passengers are defined, the destination stations of arriving passengers are specified based on a general distribution. Then the total number of passengers arrive at a station is computed according to the schedule methods. In these blocks one sever-queue system is provided for a user-specified purpose such as security checking process.

Before entering a platform, passengers may pass through several system components. In the model each system component is built as a functional block of a server-queuing model. The queue occurred at a facility results from no available severs, or the

4. MODEL DESCRIPTION
Figure 2. Example of the Passenger Flow Pattern in an APM Station Model
block or waiting to enter other facilities. The maximum number of passengers allowed to enter a facility at one time are defined as the number of servers at this facility. The process time without any queuing delay in each facility is defined by a random distribution or constant. The detail of each block is described in the block reference at the end of this chapter.

Within the interaction of the Platform block(s) and the Transit Unit block(s), the events of passenger alighting and boarding, or TU arriving and departing, can be specified. The alighting event removes passengers arriving at their destinations from TUs. The boarding event places passengers from the boarding queue in particular TUs that head to their destinations and have sufficient capacity. The door open/close time and alighting/boarding time for each passenger control the number of passengers alighting or boarding during the station dwell time. The arriving and departing events are used to determine the time the TU is dispatched as a function of the arrival time and the station dwell time. In addition, the activities of the shared platform with two guideway lanes and the separate platform with a single lane also can be simulated in the model.

After alighting, passengers leaving a station may or may not use same station facilities depending on the system configuration. At general airports, boarding and alighting passengers use the same facilities of stairway and passageway but use the different facilities of escalator and sidewalk.

4.4.2 APM Guideway Model

The APM guideway model represents the movement of the TUs on the guideway. The model contains all information about the physical guideway layout of the APM system to be simulated. The information tells what the guideway configuration is, where interlockings are located, where all switches or crossovers within them are, where the station locations are, and control strategies. The simulation selects those locations where
the important interactions and interferences between TUs, and TU and passengers occur. These locations are simulated in greater detail than other portions of this system. Most interactions and interferences occur on station platforms and interlockings such as switching, merging, and diverging points. So these are referred to as control points and have been selected to receive the most attention in the simulation. Between control points the TU movement is only simulated for energy consumption models.

Figure 3 shows the basic configurations and TU flow directions of airport APM networks constructed by the APM simulation model. The simpler network configurations include: (1) single-lane shuttle, (2) single-lane shuttle with bypass, (3) double-lane shuttle, (4) single-lane loop, (5) double-lane loop, and (6) pinched loop with turnbacks. Safe headway separation is maintained according to defined vehicle control strategies in the Veh-Schedule blocks. TUs travel along pre-defined routes. The model keeps track of the current occupancy and time integral of occupancy for all vehicles at each station, on each guideway section, and on each route of the network.

At the control point of a station platform, the passengers arrive at a station and wait in a queue until they can board a TU moving toward their desired destinations. The model logic applied in this process has been discussed in the foregoing section of APM station model.

The network is modeled by dividing each route into blocks of one guideway section from one interlocking to another. An initial TU speed of zero is assigned to each TU at a station. After departing from a station, TUs move from one guideway section of a route to another till arriving at another station, and the section capacity, cruise speed, breaking acceleration rate, and section length are specified in each section. In each section, the running speed, acceleration, travel distance, power requirements, energy consumption, occupancy and load factors are calculated by the energy consumption model.

4. MODEL DESCRIPTION
At the control point of a interlocking, the events occur. The 2-Way Switch blocks simulate the activity of a TU switched to another guideway, and are usually used in the single-shuttle with bypass network. The Merge/Diverge blocks function exactly as same as the 2-Way Swatch blocks except that there is one Merge/Diverge section allowing TUs to stop and to wait for overtaking. The Pinched Loop blocks is used in a pinched-loop guideway network. The application of these blocks are illustrated in Figure 3.

4.5 Energy Consumption Model

This section discusses how to develop the profiles of the speed, acceleration, travel time, travel distance, power requirements, energy consumption, occupancy, load factors, and LOS of a TU along the guideway route. All these profiles are determined at each increment of a pre-defined small interval of time (delta time). The model first determines the acceleration at each delta time along the route. The determination of acceleration can be related to three phases. In the first (accelerating) phase, the TU moves with a random acceleration rate calculated from the combination of tractive effort and resistance, as the traditional train simulations do. In the second (cruise) phase, the TU moves at a cruise speed as a constant speed, thus the acceleration is zero. In the final (decelerating) phase, the TU approaches a stop with a constant breaking acceleration rate. The instantaneous speed is obtained by the sum of the acceleration rate timed each delta time. It is should be noted that the TU speed is not necessary to reach the cruise speed depending on the distance between two stations. Then travel distance is calculated by the sum of each delta time timed the average speed for the interval (based the beginning speed and the ending speed). That is, the travel distance is estimated directly from the integration procedure. According to the relationships of delta time, speed, acceleration rate, and travel distance, the cumulative travel time as delay in the guideway blocks is computed as the sum of each delta time.

4. MODEL DESCRIPTION
Therefore, a smaller value of delta time can get more precise results because round off error decreases with reductions in delta time, but costs more simulation running time.

The energy consumption model built in the Energy Consumption block only calculate the energy consumption due to: (1) inherent resistance (aerodynamic + rolling resistance) and (2) gradient resistance. The major inputs required to compute the energy consumption are based on: (1) the aerodynamic characteristics of the TU and (2) the gradient of guideways. To calculate the inherent resistance the Davis’ equation is applied. In the model, the guideway route is described in terms of sections between stations, and each section is characterized by the grade (up or downhill) in percent and cruise speed. The gradient resistance is only considered as a TU goes up a hill. Therefore, the power required at each increment of delta time is obtained by the instantaneous speed timed total resistance over power efficiency. It is assumed that there is no power required when a TU moves with a braking acceleration. Finally, the total energy consumption over the route is calculated by cumulating the power required timed each delta time.

All the equations used to calculate resistances, power and energy consumption are stated in Chapter 2. The concept and process for building an APM energy consumption model are applied in the Energy Consumption block’s script as shown in Appendix D.

### 4.6 Output of the Model

A wealth of information is provided by the APM simulation model. The information could be classified as the output for each of (1) the stations and (2) the guideways.

At a station, the simulation results related to passengers and facilities are:

1. Travel time of a individual passenger
2. Average waiting time for a facility or a TU
3. Total number of passengers arriving at or leaving a station

### 4. MODEL DESCRIPTION
(4) Total number of passengers entering or leaving a facility
(5) Queue length at a facility
(6) LOS of a facility
(7) An animation display with numerical or graphical information related to passenger movements
(8) Error messages for invalid input data, invalid simulation results, and violated operations

For the single section or guideway network, the simulation results related to TUs and passengers are:

(1) Travel time of a individual TU or passenger
(2) Number of passengers in TU or vacancy, load factors, and LOS
(3) TU speed, acceleration, deceleration, and travel distance
(4) TU power requirements and energy consumption
(5) Number of TUs in a guideway or a system
(6) An animation display with numerical or graphical information related to TU maneuvers
(7) Error messages for invalid input data, invalid simulation results, and violated operations

Most results can be illustrated by graphics or tables using the APM Plotter block, Plotter Discrete Event block, Plotter Scatter block, or General Information block. The sensitivity analyses about TU speed, acceleration, deceleration, travel time, travel distance, power requirements, energy consumption, passenger loading, and load factors can be done by the APM Plotter block.

4. MODEL DESCRIPTION
4.7 Block Reference for the APM Libraries

Figure 4 shows a list of the blocks used to construct a APM station model. These blocks are in the APM Station library. Figures 7 through 18 show the structure of all the hierarchical blocks used in the station model and are arranged in alphabetical order. Figure 5 shows a list of the blocks used to build a APM guideway model. These blocks are in the APM Guideway library and the structure of all the hierarchical blocks is listed in Figures 19 through 32. The APM New Facility library contains some customized blocks that support the construction of the APM simulation model. Figure 6 shows the list of these blocks and Figures 33 through 38 show the structure of all the hierarchical blocks in this library.
Figure 4. Alphabetical List of the APM Station Library
4.7.1 Alphabetical Block Reference for the APM Station Library

1. Area of Facility Block

This block generates a constant value at each simulation step. This value represents the area of a facility and can be specified in the dialog box. This block is used for calculating the level of service at the facility.

2. Circulation Area/Passageway Block (Figure 7)

This hierarchical block represents a circulation area or a passageway facility. The walking times of passengers through the facility are set in the Input Random Number block. Also, the capacity of the facility is set in the Constant block connecting to the new activity (multiple) block. The block outputs the LOS in terms of the area/passenger.

3. Elevator Block (Figure 8)

This hierarchical block simulates a elevator activity. This block is applied for story landside terminals. The time in elevator is set in constant block. This block outputs the number of passengers taking elevator and the LOS in terms of the area/passenger.

4. MODEL DESCRIPTION
4. Entrance/Exit Block (Figure 9)

This is a hierarchical block and represents an entrance/exit facility. The facility can be defined with or without a fare collection facility. The delay for each passenger in fare collection is set in the Input Random Number block. The maximum number of passengers (capacity) through the doors is set in the Constant block. This block outputs the LOS in terms of the area/passenger.

5. Escalator Block (Figure 10)

This is a hierarchical block and represents an escalator facility. The length, speed and capacity of the escalator are set in the Constant blocks. The delay for each passenger to enter escalator is set in the Input Random Number block. This block outputs the LOS in terms of the area/passenger.

6. Pax-Assignment Block (Figure 11)

This block is a hierarchical block to assign the boarding passengers into different facilities, such as stairways, escalators, and platforms. Generally, this block is used for the APM station model which has several separate platforms for different transit lines. The assignment is based on the destination (station number) of each passenger.

4. Model Description
7. **Pax-Schedule (Method 1) Block** (Figure 12)

This block is a hierarchical block and provides the number of passengers to the facilities at discrete intervals chosen by the user. The input data of passenger demands are based on the exact arrival time and the number of passengers which are set in the Program block. In addition, the destination (station number) of each passenger should be assigned.

8. **Pax-Schedule (Method 2) Block** (Figure 13)

This block is a hierarchical block and provides the number of passengers to the facilities at discrete intervals chosen by the user. The input data of passenger demands are based on the passenger arrival rate (pax/sec) and corresponding time (sec). This method provides a straight line approximation of input curves. The data are set in the generator block. The destination (station number) of all passengers is assigned in the Input Random Number block.

9. **Pax-Schedule (Method 3) Block** (Figure 14)

This hierarchical block provides the number of passengers to the facilities at a random probability distribution chosen by the user. This method provides passenger arrivals which are modeled as variables with randomly distributed interarrival times. The data are set in the generator block. The destination (station number) of each passenger is based on a general distribution.

4. **MODEL DESCRIPTION**
10. Platform Block (Figure 15)

This is a hierarchical block to represents a platform facility. The walking times of boarding passengers on platform to catch the vehicle or debarking passengers to leave platform are set in the Input Random Number block. The capacity of the facility is set in the Constant block. The B1 connector is used to output the boarding passengers taking the eastbound transit unit, and the B2 connector is used to output the boarding passengers taking the westbound flow. The block outputs the LOS in terms of the area/passenger and average waiting time per boarding passenger.

10. Sidewalk Block (Figure 16)

This is a hierarchical block and represents a sidewalk facility. The length, speed and capacity of the sidewalk are set in the Constant blocks. The delay for each passenger to enter sidewalk is set in the Input Random Number block. This block outputs the LOS in terms of the area/passenger.

12. Stairway Block (Figure 17)

This hierarchical block represents a stairway facility. The area and capacity of the stairway are set in the Constant blocks. The walking times of passengers on stairway are set in the Input Random Number block. Also, the user should set the maximum number of passengers allowed to enter stairway at one time. This block outputs the LOS in terms of the area/passenger.

4. MODEL DESCRIPTION
13. Station with Double Lanes Block

This block represents a station facility with a double-lane guideway system. All APM station blocks required should be built inside this block for completing the whole APM system.

14. Station with Single Lane Block

This block represents a station facility with a single-lane guideway system. All APM station blocks required should be built inside this block for completing the whole APM system.

15. Transit Unit Block (Figure 18)

This hierarchical block to represent passenger and vehicle activities at a station. This block is connected with the platform to simulate passengers boarding and alighting and transit units arriving and departing. The basic input data of a transit unit include number of vehicles, capacity of one vehicle, total number of doors, and door opening time. Other input parameters are average boarding and debarking time, station dwell time, and area of the transit unit. The B1 connector receives the boarding passenger flow taking the eastbound transit unit, and the B2 connector receives the boarding passenger flow taking the westbound flow. The D connector outputs the passenger flow alighting from a TU.

4. MODEL DESCRIPTION
Figure 5. Alphabetical List of the APM Guideway Library
4.7.2 Alphabetical Block Reference for the APM Guideway Library

1. 2-Way Switch (1) Block (Figure 19)
   This is a hierarchical block to represent a 2-way switch guideway. This block can be used to construct a bypass guideway system, connect the Station with Double Lanes block, or be a merge / diverge guideway. All transit unit numbers and indicators should be set in the Conversion Table Block for switching guideway. The indicators represent whether transit units switch from point A to point B or to point C.

2. 2-Way Switch (2) Block (Figure 20)
   This block functions exactly as same as the 2-Way Switch (1) block.

3. Merge/Diverge (1) Block (Figure 21)
   This hierarchical block represents a merge / diverge guideway. This block can be used to connect the Station with Single Lane block as an off-line station, be a simple merge / diverge guideway or be a bypass guideway system. In this block, the transit unit traveling from point B to point A can be limited to enter the main guideway (A-C) until one has passed point A. All transit unit numbers and indicators should be set in the Conversion Table block for switching guideway. The indicators represent whether transit units switch.

4. MODEL DESCRIPTION
from point A to point B or to point C. In addition, the distance between A and B, and speed of transit units are set in the Constant blocks.

4. Merge/Diverge (2) Block (Figure 22)

This block functions exactly as the Merge / Diverge (1) block.

5. Merge/Diverge (3) Block (Figure 23)

This block functions exactly as the Merge / Diverge (1) block.

6. Merge/Diverge (4) Block (Figure 24)

This block functions exactly as the Merge / Diverge (1) block.

7. Pinched Loop (1) Block (Figure 25)

This hierarchical block is used in a pinched-loop guideway system. All transit unit numbers and indicators should be set in the Conversion Table blocks for switching guideway from A to B or from B to A. The indicators represent whether transit units switch
to the other guideway or not. In addition, the distance between A and B, and speed of transit units are set in the Constant blocks.

8. Pinched Loop (2) Block (Figure 26)

This block functions exactly as the Pinched Loop (1) block.

9. Single-Lane (1) Block (Figure 27)

This hierarchical block represents a simple single-lane guideway. The travel time of a transit unit in this guideway depends on the distance of this guideway, average speed, and if necessary, the guideway and station safety headways. These parameters are set in the Constant blocks. Also, the user can limit the number of transit units that can be in this guideway section. The animation shows the TU traveling in this section and it’s TU number.

10. Single-Lane (2) Block (Figure 27)

This block functions exactly as the Single-Lane (1) block.

11. Single-Lane (3) Block (Figure 27)

This block functions exactly as the Single-Lane (1) block.

4. MODEL DESCRIPTION
12. **Single-Lane (4) Block** (Figure 27)

This block functions exactly as the Single-Lane (1) block.

13. **Single-Lane (5) - Loop Block** (Figure 28)

This block functions similarly as the Single-Lane (1) block but it is used only in the loop guideway configuration.

14. **Single-Lane (6) - Loop Block** (Figure 29)

This block functions similarly as the Single-Lane (1) block but it is used only in the loop guideway configuration.

15. **Turnaround (1) Block** (Figure 30)

This hierarchical block is used to change transit units traveling in a eastbound direction to a westbound direction.
16. **Turnaround (2) Block** (Figure 31)

This hierarchical block is used to change transit units traveling in a westbound direction to a eastbound direction.

17. **Veh-Schedule Block** (Figure 32)

This hierarchical block provides the number of transit units to the APM system at discrete intervals chosen by the user. The input data of vehicle demands are based on the exact departure time and the number of transit units which are set in the Program block. In addition, the transit unit number should be assigned for each schedule.

---

4. **MODEL DESCRIPTION**
Figure 6. Alphabetical List of the APM New Utility Library
4.7.3 Alphabetical Block Reference for the APM New Utility Library

1. APM Energy Data Block

This block can be used to input the data as global variables used for the APM simulation. The data related to the physical characteristics of TUs and some parameters include: (1) mass of a TU, (2) number of axles per TU, (3) average load per TU, (4) power efficiency, (5) cross sectional area, (6) experimental coefficient, (7) drag coefficient, (8) delta time for integration, (9) number of vehicles per TU, (10) capacity of one vehicle, (11) number of effective doors, (12) door open/close time, (13) area per vehicle, and (14) average boarding/debarking time. The block is applied in the Veh-Schedule hierarchical block.

2. APM Plotter Block (Figure 33)

This hierarchical block can be used to give eight figures from a data file and tabulate data. These twelve figures are acceleration vs. time, acceleration vs. distance, velocity vs. time, velocity vs. distance, power vs. time, power vs. distance, energy vs. time, energy vs. distance, load factor vs. distance, occupancy vs. distance, distance vs. time, and LOS vs. distance figures. This block consists of four New File Input blocks and twelve Plotter, Scatter (4) blocks. This block should be executed separately in a continuous simulation after completing the APM model simulation. To read all data, the simulation time and the input value of max. rows must be greater than the row number of data.

4. MODEL DESCRIPTION
3. Arrival Rate Block
This block is only used in the Pax-Schedule (method 2) hierarchical block. It is connected to the input data block. This block converts the passenger arrival rates specified in the input data block into the number of passengers arriving at discrete intervals chosen by the user. The T connector outputs the total number of passengers arriving.

4. Boarding/Alighting Block (Figure 34)
This block is applied in the Transit Unit hierarchical block to simulate the process of passenger boarding and alighting. The value through the Y connector is to determine the boarding events. The D connector stands for the direction of a TU arriving at a station. The station dwell time and number of TUs at a station are specified through the S and T connectors. The N connectors are specified by the TU number of a TU arriving at a station. The five item input connectors and three item output connectors are linked with Platform blocks and guideway blocks in order to continue the flow of passengers through a platform facility to a TU.

5. Distributor (1) Block (Figure 35)
This hierarchical block passes items from two different sources into parallel process activities based on a general random distribution. The probability for sending items through one of output connectors is specified in the Input Random Number block. This block is

4. MODEL DESCRIPTION
used in APM station model for passengers to take different facilities such as stairway, elevator, and escalator. In the APM station model, this block is used to send boarding passengers to different facilities.

6. Distributor (2) Block (Figure 36)

This block functions as same as the Distributor (1) block. In the APM station model, it is only used to pass a debarking passenger flow into different facilities.

7. Energy Consumption Block

This block is used to simulate the energy consumption model to calculate the TU’s travel time, speed, acceleration, power requirements, energy consumption, occupancy, and load factors. The Energy Consumption block is located in the Guideway Maneuver Block of each guideway block to complete an APM guideway model. The results of this model will be written into a data file defined in the Veh-Schedule model. The block’s script is shown in Appendix D. The initial values of the TU’s number, passenger occupancy, TU direction, velocity, acceleration, energy consumption, power requirements, travel distance are specified respectively in the N, X, D, S, A, E, P, and L connectors. The initial values of the TU’s number, passenger occupancy, TU direction, velocity, acceleration, energy consumption, power requirements, travel distance are specified respectively in the N, X, D, S, A, E, P, and L input connectors. This block also outputs the results of the TU’s velocity, acceleration, energy consumption, power requirements, and travel distance through S, A, E, P, and L output connectors over this guideway section.

4. MODEL DESCRIPTION
8. General Information Block (Figure 37)

This hierarchical block counts the number of items and displays information about the items that pass through it. In the APM simulation model, the typical information based on different purposes are shown in this block: origins and destinations of passengers, arrival/departure time of transit units, travel time of passengers and transit units, transit unit number, and total items entering or leaving the facility with respect to time. All information is shown in the Information block and the Plotter, Discrete Event block.

9. Guideway Maneuver Block (Figure 38)

This hierarchical block consisting of the Energy Consumption block is used to simulate the TU’s movement along a guideway section. So the block is a part of all guideway blocks. This block displays and provides data for conducting energy consumption model.

10. New Activity, Delay Block

This block is identical to the Activity, Delay block of the Discrete-Event library except that it outputs the total number of items passing and the number of items held. This block holds only one item for a specified amount of time, then releases it. The holding time is the value in the dialog or, if connected, the value at the D connector when the item is received (the connector overrides the dialog).
11. New Activity, Multiple Block

This block is same to the Activity, Multiple block of the Discrete-Event library except that it can output the total number of items passing and there is one input connector to set the maximum number (capacity) of items held. This block holds many items and passes them out based on the delay and arrival time for each item. The item with the smallest total delay and earliest arrival time is passed out first. The delay time for each item is set through the D connector or, if nothing is connected there, can be specified in the dialog. In the APM station model, this block can be used to represent vehicle doors and limit the maximum number of passengers getting in or getting off a TU at one time.

12. New Add Block

This block is used for basic math purposes. It outputs the total value by adding the all input values on the left of the block.

13. New Combine (1) Block

This block is similar to the Combine block of the Discrete-Event library and differs only in the number of input connectors available. This block combines the items from different blocks into a single stream of items. This block does not change any properties of input items and are not batched together. This block is used to pass boarding passengers into a specific facility in the APM station model.

4. Model Description
14. New Combine (2) Block

This block functions exactly as the New Combine (1) block but it is only used to send debarking passengers to a specific facility in the APM station model.

15. New Combine (3) Block

This block functions exactly as the Combine block of the Discrete-Event library to combine the items from two different sources into a single stream of items.

16. New Equation Block

This block is used for basic math purposes and delivers error messages for checking APM system variables such as the sufficiency of dwell time for debarking passengers. This block allows to enter a free-form equation into the dialog and Extend will calculate the results. The user can apply Extend's built-in operators and functions, and some or all of the input values as part of the equation.

17. New File Input Block

This block reads data from a data file and writes it into the block's table in dialog. Once the data is in the table, it can be used in model. The user selects the file to be read by
typing its path name, or can leave the name blank so Extend prompts the user for file name and fills in the path name. If a path name is entered here, the user can specify folders by using colons such as "My Disk:Data folder:Data filename". The file can have up to nine columns of real data including travel time, acceleration, speed, travel distance, power required, energy required, occupancy, load factor, and LOS. All these data are created by the energy consumption block. The file can have as many rows as the user wants, but the user must specify the maximum number of rows that he or she wants to read from the file in the dialog. Table values in each column can be used in the simulation. The left output connector has the values in the first column, the right output connector has the values in the ninth column. This block is used in the APM Plotter block.

18. New Prioritizer Block

This block is similar to the Prioritizer block of the Discrete-Event library. The detail description of this block is referred to the Prioritizer block. This block differs from the Prioritizer block if the corresponding output is not connected to this block, the dialog will show a non-zero default value instead of a blank value for the dialog once the simulation is run. Then, the different simulation runs can be continued without setting the priority to override the blank values in the dialog again.

19. New Queue, FIFO Block

This block provides a first-in-first-out (FIFO) queue and is identical to the Queue, FIFO block of the discrete-event library. The maximum queue length, which determines

4. MODEL DESCRIPTION
how many items the queue can hold, can be set in the dialog. The user can specify that the simulation should stop when the queue is full (reaches the maximum length). The # connector outputs the total number of items passing. In addition, the block outputs the average queue length, average wait time, and utilization of the queue through output connectors. This block is used for calculating the average waiting time per boarding passenger in the APM station model.

20. New Select Output (5) Block

The block passes the input value to one of five outputs according to the value of the T connector. The top output is selected if the T connector is 1 and the bottom output is selected if it is 5. For connectors that are not selected, the dialog lets the user specify an output value of either noValue, 0, or a repeat of the last value they output. This block was modified from the select output block of the generic library and is only used to construct the Distributor (1) and the Distributor (2) blocks.
4. MODEL DESCRIPTION

**Input data**

Area of facility: 1000

Capacity of facility: 200

Max. no. of pax entering circulation area / passageway at one time: 10

Capacity of passageway: 200

Time in walking on circulation area / passageway:

1) Min = 20

2) Max = 40

Time of passengers in walking to other facilities

Max. number of pax entering circulation area / passageway at one time

Figure 7. Structure of the Circulation Area/Passageway Block (APM station Library)
4. MODEL DESCRIPTION

Input data

Capacity of elevator: 10
Area of facility: 50
Time to enter / exit elevator:
   (random distribution)
   1) Min = 0
   2) Max = 3
   Time in elevator: 50

No. of pax taking elevator

Elevator for 2 floors only

Figure 8. Structure of the Elevator Block (APM station Library)
Figure 9. Structure of the Entrance/Exit Block (APM station Library)
Input data

Length of escalator: 30
Speed of escalator: 5
Capacity of escalator: 30

Area of facility: 98

Delay to enter escalator:
1) Min = 0
2) Max = 2

Figure 10. Structure of the Escalator Block (APM station Library)
4. MODEL DESCRIPTION

Input Data

Input the data for assigning Pax into different facilities based on their destination stations:

<table>
<thead>
<tr>
<th>Row</th>
<th>x_in</th>
<th>y_out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>2</td>
</tr>
</tbody>
</table>

Figure 11. Structure of the Pax-Assignment Block (APM station Library)
**Input data**

**Method (1):** The input data of pax demand are based on the exact arrival time and the number of pax. For example, the pax transferring from international terminal to domestic terminal take the APM system directly after debarking from international flights. Therefore, the arrival time of pax is same to the flight arrival time.

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>(No. of Pax)</th>
<th>(Destination) Station Number</th>
<th>Value</th>
<th>Priority</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1800</td>
<td>172</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1800</td>
<td>139</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>1200</td>
<td>120</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1800</td>
<td>172</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2100</td>
<td>139</td>
<td>7</td>
<td></td>
</tr>
</tbody>
</table>

**Considerations:** (if necessary)

- Service processing time/delay per pax:
  - **Min = 0**
  - **Max = 600**
- Number of servers: **2000**

**Possible destination stations:** If using this function, insert "1" else "0".

<table>
<thead>
<tr>
<th>Row</th>
<th>Value</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 12. Structure of the Pax-Schedule (Method 1) Block (APM station Library)**
**Input data**

**Method (2):** The input data of pax demand are based on the pax arrival rate (pax/sec) and corresponding time (sec). This method provides a straight line approximation of the input curves.

**Output is:** ☑ stepped ☐ interpolated

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time</td>
<td>Y Output</td>
</tr>
<tr>
<td>---</td>
<td>-----</td>
<td>----------</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0.125</td>
</tr>
<tr>
<td>1</td>
<td>600</td>
<td>0.125</td>
</tr>
<tr>
<td>2</td>
<td>800</td>
<td>0.5</td>
</tr>
<tr>
<td>3</td>
<td>2400</td>
<td>3.5</td>
</tr>
</tbody>
</table>

**Considerations:** (if necessary)

- Service processing time/delay per pax (random distribution)
  - 1) Min = \(\emptyset\)
  - 2) Max = \(\emptyset\)

**Number of servers:** 1000

**Possible destination stations:**

<table>
<thead>
<tr>
<th>Row</th>
<th>Value</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>100</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Figure 13. Structure of the Pax-Schedule (Method 2) Block (APM station Library)**
4. MODEL DESCRIPTION

Input data

Method (3): The input data of pax demand are based on a random distribution as shown as follows. The destination of each pax is based on a general distribution on right. This method provides pax arrivals which are modeled as variables with randomly distributed interarrival times.

- Binomial
- Constant
- Erlang
- Exponential
- HyperExponential
- Integer, uniform
- LogNormal
- Normal
- Poisson
- Real, uniform
- Triangular: most likely value
- Weibull

(1) Mean = 10
(2) Unused = 0

Considerations (if necessary)

Service processing time/delay per pax:
(1) Min = 0
(2) Max = 0

Number of servers: 1000

Possible destination stations:

<table>
<thead>
<tr>
<th>Row</th>
<th>Value</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>2</td>
<td>50</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>50</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>50</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>50</td>
</tr>
<tr>
<td>4</td>
<td>6</td>
<td>50</td>
</tr>
<tr>
<td>5</td>
<td>7</td>
<td>50</td>
</tr>
</tbody>
</table>

Total number of arriving pax

Figure 14. Structure of the Pax-Schedule (Method 3) Block (APM station Library)
**Input data**

Station number: 2

The capacity of platform: 700

Area of facility (platform area): 350

Time for boarding pax to enter vehicle or for deboarding pax to leave platform without queueing and waiting:

1) \( \text{Min} = 5 \)
2) \( \text{Max} = 10 \)

Stations that TUs go forward (eastbound) or backward (westbound):

(Direction: forward/eastbound => 1; backward/westbound => 0)

<table>
<thead>
<tr>
<th>Station No.</th>
<th>Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
</tr>
</tbody>
</table>

**Figure 15. Structure of the Platform Block (APM station Library)**
Figure 16. Structure of the Sidewalk Block (APM station Library)
**Input data**

- Capacity of stairway: 50
- Max. no. of pax allowed to enter stairway at one time: 3
- Assume time for first step of pax on stairway: 1
- Area of facility (stairway area): 250
- Time in walking on stairway:
  1. Min = 10
  2. Max = 20

**Figure 17. Structure of the Stairway Block (APM station Library)**
**Input data**

Station number: 2

TUs leaving station reverse direction or not: 0

Station dwell time: 35

---

**Figure 18. Structure of the Transit Unit Block (APM station Library)**
**Input Data**

Input the TU number and the indicator for switching guideway:
(Y=0 means no switch; Y=1 means switch)

(1) For TUs which switch from A to B:
(TU Number) (Switch/Unswitch)

<table>
<thead>
<tr>
<th>Row</th>
<th>x_in</th>
<th>y_out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 19. Structure of the 2-Way Switch (1) Block (APM Guideway Library)
Input Data

Input the TU number and the indicator for switching guideway:
(Y=0 means no switch; Y=1 means switch)

(1) For TUs which switch from A to B:
(TU Number) (Switch/Unswitch)

<table>
<thead>
<tr>
<th>Row</th>
<th>x in</th>
<th>y out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 20. Structure of the 2-Way Switch (2) Block (APM Guideway Library)
Figure 21. Structure of the Merge/Diverge (1) Block (APM Guideway Library)
Input Data

Distance between A and B on this section: 200
Distance to the eastern station: 0
Distance to the western station: 0
Cruise Speed: 10
Breaking acceleration: -1
Gradient of this guideway: 0
Max. number of TUs on this guideway: 2
Does TU at B should stop until one on the main guideway has passed A: 1

Input the TU number and the indicator for switching guideway: (Y=0 means no switch; Y=1 means switch)

1) For TUs which switch from A to B:
   (TU Number) (Switch/Unswitch)

<< Speed/Tractive force table >>

Figure 22. Structure of the Merge/Diverge (2) Block (APM Guideway Library)
Figure 23. Structure of the Merge/Diverge (3) Block (APM Guideway Library)
Input Data

Distance between A and B on this section: 200
Distance to the eastern station: 0
Distance to the western station: 0
Cruise Speed: 10
Breaking acceleration: -1
Gradient of this guideway: 0
Max. number of TUs on this guideway: 2
Does TU at B should stop until one on the main guideway has passed A?: (1 => Yes; 0 => No) 1

Input the TU number and the indicator for switching guideway: (Y=0 means no switch; Y=1 means switch)

(1) For TUs which switch from A to B:
(TU Number) (Switch/Unswitch)
Row | x_in | y_out |
--- | --- | --- |
0   | 1   | 0    |
1   | 2   | 0    |
2   | 3   | 0    |

<< Speed/Ttractive force table >>

<table>
<thead>
<tr>
<th>Row</th>
<th>Speed (m/s)</th>
<th>Ttractive (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>70000</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>58000</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>40000</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>30000</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>15000</td>
</tr>
</tbody>
</table>

Figure 24. Structure of the Merge/Diverge (4) Block (APM Guideway Library)
**Input Data**

Distance between A and B on this section: 200
Distance to the eastern station: 0
Distance to the western station: 0
Cruise speed: 10
Breaking acceleration: -1
Gradient of this section: 0

Max. number of TUs on this guideway: 2

<< Speed/Tractive force table >>

<table>
<thead>
<tr>
<th>Row</th>
<th>Speed (m/s)</th>
<th>Tractive (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>70000</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>55000</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>40000</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>30000</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>15000</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>5000</td>
</tr>
</tbody>
</table>

Input the TU number and the indicator for switching guideway (Y=0 means no switch; Y=1 means switch):

(1) For TUs which switch from A to B:
(TU Number) (Switch/Unswitch)

<table>
<thead>
<tr>
<th>Row</th>
<th>x_in</th>
<th>y_out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

(2) For TUs which switch from B to A:
(TU Number) (Switch/Unswitch)

<table>
<thead>
<tr>
<th>Row</th>
<th>x_in</th>
<th>y_out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

---

Figure 25. Structure of the Pinched Loop (1) Block (APM Guideway Library)
**Input Data**

- Distance between A and B on this section: **200**
- Distance to the eastern station: **0**
- Distance to the western station: **0**
- Cruise speed: **10**
- Breaking acceleration: **-1**
- Gradient of this section: **0**

Max. number of TUs on this guideway: **2**

---

**<< Speed/Tractive force table >>**

<table>
<thead>
<tr>
<th>Row</th>
<th>Speed (m/s)</th>
<th>Tractive (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>70000</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>50000</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>40000</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>30000</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>15000</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>5000</td>
</tr>
</tbody>
</table>

---

Input the TU number and the indicator for switching guideway (Y=0 means no switch; Y=1 means switch):

(1) For TUs which switch from A to B:

<table>
<thead>
<tr>
<th>Row</th>
<th>x in</th>
<th>y out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

(2) For TUs which switch from B to A:

<table>
<thead>
<tr>
<th>Row</th>
<th>x in</th>
<th>y out</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0</td>
</tr>
</tbody>
</table>

---

Figure 26. Structure of the Pinched Loop (2) Block (APM Guideway Library)
Input Data

Length of this guideway: 200
Distance to the eastern station: 0
Distance to the western station: 0
Cruise speed: 10
Breaking acceleration: -1
Gradient of this guideway:
(positive is for going up eastbound and negative is for going down eastbound)
Max. number of TUs in this guideway: 2
Current TU in this guideway (TU No.): 1

<< Speed/Tractive force table >>

<table>
<thead>
<tr>
<th>Row</th>
<th>Speed (m/s)</th>
<th>Tractive (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>70000</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>55000</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>40000</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>30000</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>15000</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>5000</td>
</tr>
<tr>
<td>6</td>
<td>60</td>
<td>2500</td>
</tr>
<tr>
<td>7</td>
<td>70</td>
<td>1000</td>
</tr>
</tbody>
</table>

Travel time (display only): 25.7454835802
Distance traveled (position): 1400
No. of pax in TU: 0

Figure 27. Structure of the Single-Lane (1), (2), (3) or (4) Block (APM Guideway Library)
4. MODEL DESCRIPTION

**Input Data**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Length of this guideway:</td>
<td>200</td>
</tr>
<tr>
<td>Distance from station to connect A:</td>
<td>0</td>
</tr>
<tr>
<td>Distance from station to connect B:</td>
<td>0</td>
</tr>
<tr>
<td>Cruise speed:</td>
<td>10</td>
</tr>
<tr>
<td>Breaking acceleration:</td>
<td>-1</td>
</tr>
<tr>
<td>Gradient of this guideway: (positive is for going up eastbound and negative is for going down eastbound)</td>
<td>0</td>
</tr>
<tr>
<td>Max. number of TUs on this guideway:</td>
<td>2</td>
</tr>
<tr>
<td>Current TU on this guideway (TU No.):</td>
<td>1</td>
</tr>
</tbody>
</table>

**<< Speed/Tractive force table >>**

<table>
<thead>
<tr>
<th>Row</th>
<th>Speed (m/s)</th>
<th>Tractive (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>70000</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>55000</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>40000</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>30000</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>15000</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>5000</td>
</tr>
<tr>
<td>6</td>
<td>60</td>
<td>2500</td>
</tr>
<tr>
<td>7</td>
<td>70</td>
<td>1000</td>
</tr>
</tbody>
</table>

Travel time (display only): 25.7454835802
Distance traveled (position): 0
No. of Pax in TU:

![Diagram of model description and simulation setup](image)

**Figure 28. Structure of the Single-Lane (5) Block (APM Guideway Library)**
Input Data

Length of this guideway : 200
Distance from station to connect A : 0
Distance from station to connect B : 0
Cruise speed : 10
Breaking acceleration : -1
Gradient of this guideway :
(positive is for going up eastbound and negative is for going down eastbound)
Max. number of TUs on this guideway : 2
Current TU on this guideway (TU No.) : 1

<< Speed/Tractive force table >>

<table>
<thead>
<tr>
<th>Row</th>
<th>Speed (m/s)</th>
<th>Tractive (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>75000</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>55000</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>40000</td>
</tr>
<tr>
<td>3</td>
<td>30</td>
<td>30000</td>
</tr>
<tr>
<td>4</td>
<td>40</td>
<td>15000</td>
</tr>
<tr>
<td>5</td>
<td>50</td>
<td>5000</td>
</tr>
<tr>
<td>6</td>
<td>60</td>
<td>2500</td>
</tr>
<tr>
<td>7</td>
<td>70</td>
<td>1000</td>
</tr>
</tbody>
</table>

Travel time (display only) : 25.7454835882
Distance traveled (position) : 0
No. of pax in TU :

Figure 29. Structure of the Single-Lane (6) Block (APM Guideway Library)
Input Data

Average turnaround time per TU: **30**

Figure 30. Structure of the Turnaround (1) Block (APM Guideway Library)
4. MODEL DESCRIPTION

Input Data

Average turnaround time per TU: 20

Figure 31. Structure of the Turnaround (2) Block (APM Guideway Library)
Figure 32. Structure of the Veh-Schedule Block (APM Guideway Library)
Figure 33. Structure of the APM Plotter Block (APM New Utility Library)
Input data

Build a general distribution:

- **General**  - **Discrete**

**General values**:

<table>
<thead>
<tr>
<th>Row</th>
<th>Value</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0.475</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0.475</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0.08</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0.02</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

Figure 35. Structure of the Distributor (1) Block (APM New Utility Library)
Input data

Build a general distribution:

- General  ○ Discrete

**General values =**

<table>
<thead>
<tr>
<th>Row</th>
<th>Value</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0.5</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0.3</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>0.2</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>0.0</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Figure 36. Structure of the Distributor (2) Block (APM New Utility Library)
4. MODEL DESCRIPTION

**Input Data**

Does this block pass items out of the simulation?
(Yes => 1; No => 0)

(1) Top : 1
(2) Bottom : 0

Total items enter or leave facility

Figure 37. Structure of the General Information Block (APM New Utility Library)
**Data Display**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mass of TU</td>
<td>70000</td>
</tr>
<tr>
<td>Number of axles</td>
<td>6</td>
</tr>
<tr>
<td>Average load per axles</td>
<td>11667</td>
</tr>
<tr>
<td>Cross sectional area (A)</td>
<td>9</td>
</tr>
<tr>
<td>Experimental coefficient (B)</td>
<td>0.03</td>
</tr>
<tr>
<td>Drag coefficient (C)</td>
<td>0.00034</td>
</tr>
<tr>
<td>Power efficient</td>
<td>0.85</td>
</tr>
<tr>
<td>Delta time for integration</td>
<td>1</td>
</tr>
<tr>
<td>TU number</td>
<td>1</td>
</tr>
<tr>
<td>TU direction</td>
<td>1</td>
</tr>
<tr>
<td>Current time</td>
<td>0</td>
</tr>
<tr>
<td>Initial speed of TU</td>
<td>0</td>
</tr>
<tr>
<td>Current acceleration</td>
<td>0</td>
</tr>
<tr>
<td>Current speed</td>
<td>0</td>
</tr>
<tr>
<td>Distance traveled</td>
<td>0</td>
</tr>
<tr>
<td>Breaking distance</td>
<td>50</td>
</tr>
<tr>
<td>No. of pax in TU</td>
<td>0</td>
</tr>
<tr>
<td>Load factor</td>
<td>0</td>
</tr>
<tr>
<td>Travel time</td>
<td>25.7454835802</td>
</tr>
<tr>
<td>Number of arrivals</td>
<td>0</td>
</tr>
<tr>
<td>Number of departures</td>
<td>0</td>
</tr>
<tr>
<td>Power required</td>
<td></td>
</tr>
<tr>
<td>Energy consumed</td>
<td></td>
</tr>
<tr>
<td>Utilization</td>
<td>0</td>
</tr>
</tbody>
</table>

**Figure 38. Structure of the Guideway Maneuver Block (APM New Utility Library)**
5. MODEL APPLICATION AND ANALYSIS

5.1 Introduction

This chapter presents the application of the APM simulation model developed. The APM system at William B. Hartsfield Atlanta International Airport (ATL) was simulated to illustrate the use of the model. The APM station models and APM guideway models were tested and analyzed for a base year (1995) and horizon year (2010) demand levels.

5.2 Model Application

At William B. Hartsfield Atlanta International Airport (ATL) total passenger traffic in 1994 reached a record 54 million with 31 domestic and international airlines servicing over 200 destinations [Airport Commissioner’s Office, 1995]. The passenger terminal complex area includes two terminal buildings, one international and five domestic concourses (24 international and 158 domestic gates) connected by a 2800 m. underground transit mall. In this research the APM simulation model was applied at ATL to simulate the APM system operations. The APM configuration at ATL was constructed as pinched-loop system and is shown in Figure 39. Normal traffic is counter-clockwise from the south Ticketing station to south Concourses A, B, C, D and E. The TUs leaving Concourse E stop behind the switch, reverse direction, cross over to the north guideway, and travel to E, D, C, B, A, Ticketing station, and Baggage station. The TUs then switch back to the south guideway and repeat the cycle.
Figure 39. ATL APM System Configuration
5.2.1 Model Assumptions

To demonstrate the capabilities of the model, the assuming data for model inputs can be categorized as follows:

I. *Vehicle characteristics*

1. There are seven separate TUs of three vehicles operating in the system during simulation time.

2. Each vehicle has two doors opened at each station stop. The door open/close time is 1.5 second. Average passenger loading/unloading rate is 1 s/passenger.

3. Each vehicle can hold 65 passengers. The floor area of vehicle is 25 m².

4. The weight of vehicle is 70000 Newtons. Each vehicle has 2 axles and its cross sectional area is 9 m².

5. Power efficient is 0.85. Experimental coefficient is 0.03 and drag coefficient is 0.00034.

II. *Network*

1. Average cruise speed on guideway is 10 m/s and breaking acceleration is 1 m/s.

2. The guideway dimensions are referred to Figure 40 as a APM guideway model. The guideway gradient is +0.2 % from Baggage station to Concourse E.

3. The station dwell times are 35 seconds for all stations.

4. The minimum safety headway is 120 seconds.

III. *Station characteristics*

1. At Baggage station there are one shared platform, two escalators for going up and one for going down, one elevator and one stairway (Figure 41).
2. At Ticketing station there are one shared platform, two escalators for going down and one for going up, one elevator and one stairway (Figure 41).

3. At each Concourses A, B, C and D station there are two separate platforms, two pairs of escalators, one elevator and one stairway for boarding and alighting passengers (Figure 41).

4. At Concourse E station there are one shared platform, two escalators for going up and one for going down, one elevator, and one stairway (Figure 41).

5. The area for each station platform is 355 m$^2$. The escalator dimensions are: width 1.2 m, length 30 m. Their speed is 0.5 m/s. The trip time in elevator is 10 seconds and area is 6 m$^2$. The area for each stairway is 90 m$^2$.

6. The probability of passengers entering platforms by taking escalators is 95%, by taking elevators is 2%, and by taking stairway is 3%.

7. It is assumed that 100% of all passengers will use the APM system facility when they enter or leave terminals and concourses.

5.2.2 Base Year Scenario

In 1994, total passenger traffic of 54 million placed Atlanta in second position in the world, behind Chicago O'Hare Airport, in terms of the average daily operations (takeoffs and landings) [Airport Commissioner's Office, 1995]. Airlines significantly contributing to Atlanta's passenger increase were Delta, ValuJet, Continental and Kiwi International. Today, fifty percent of the airport's passengers originate or end in Atlanta. In the model the typical weekday busy hour arrivals are listed in Table 10 based on Official Airline Guides [OAG, Aug. 31, 1995]. It is assumed that the probability is 0.5 that deplanements are transfers and the connecting concourse of each passenger is based on a random distribution. All enplaning passengers start arriving at the APM station between 30 and 20
minutes before departure time. All information related to the APM passenger demand flows are shown on Table 10. The busy hours occur from 11:30 AM to 1:00 PM with a total of 88 flights arriving and the simulation time was set from 12:00 PM to 1:00 PM.

5.2.3 Horizon Year Scenario

The demand for horizon year (2010) is assumed to reflect a 30% increase from that of the base year. The service facilities are assumed to remain the same as in the base year scenario.

5.3 Model Results

Some important and representative results of the ALT APM simulation model are shown in Figures 42 through 63. Figures 42 through 47 show the relationships of travel distance, acceleration, velocity, and simulation time for the particular vehicle of TU No.4. Figures 48 through 50 show the power required and energy consumed by TU No.4. In addition, load factors, occupancies and LOS of TU No.4 in a particular route for the base and horizon year scenarios are illustrated in Figures 51 through 53. The equivalent LOS of a facility can be defined as the average LOS for total number of passengers. Figures 54 through 61 show the average waiting times, passenger queuing, and LOS measures of some system facilities at Concourse A. According to the Table 3 of LOS standards, the LOS of a facility can be easily defined. Figures 62 and 63 show the number of passengers arriving at Concourse A Station by APM system and taking the escalator. These figures only represent some important operational characteristics for particular facilities. However, the analysis of an airport APM system is accomplished based on comprehensive evaluation and reliable results.

5. MODEL APPLICATION AND ANALYSIS
### Table 10. Flight Schedule during Busy Hours at ATL Airport

[Official Airline Guides, Aug. 31, 1995]

<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>Airline</th>
<th>Concourse</th>
<th>Departure Time</th>
<th>Aircraft Type</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>11:30 AM</td>
<td>DL</td>
<td>A</td>
<td>12:30 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>11:44 AM</td>
<td>DL</td>
<td>A</td>
<td>12:44 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
<tr>
<td>11:50 AM</td>
<td>DL</td>
<td>A</td>
<td>12:50 PM</td>
<td>L10</td>
<td>256</td>
</tr>
<tr>
<td>12:07 PM</td>
<td>DL</td>
<td>A</td>
<td>1:07 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:12 PM</td>
<td>DL</td>
<td>A</td>
<td>1:12 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:13 PM</td>
<td>DL</td>
<td>A</td>
<td>1:13 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:15 PM</td>
<td>DL</td>
<td>A</td>
<td>1:15 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
<tr>
<td>12:20 PM</td>
<td>DL</td>
<td>A</td>
<td>1:20 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
<tr>
<td>12:20 PM</td>
<td>DL</td>
<td>A</td>
<td>1:20 PM</td>
<td>763</td>
<td>216</td>
</tr>
<tr>
<td>12:28 PM</td>
<td>DL</td>
<td>A</td>
<td>1:28 PM</td>
<td>757</td>
<td>178</td>
</tr>
<tr>
<td>12:30 PM</td>
<td>DL</td>
<td>A</td>
<td>1:30 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:31 PM</td>
<td>DL</td>
<td>A</td>
<td>1:30 PM</td>
<td>L10</td>
<td>256</td>
</tr>
<tr>
<td>12:32 PM</td>
<td>DL</td>
<td>A</td>
<td>1:32 PM</td>
<td>73S</td>
<td>120</td>
</tr>
<tr>
<td>12:33 PM</td>
<td>DL</td>
<td>A</td>
<td>1:33 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:35 PM</td>
<td>DL</td>
<td>A</td>
<td>1:35 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
<tr>
<td>12:37 PM</td>
<td>DL</td>
<td>A</td>
<td>1:37 PM</td>
<td>757</td>
<td>178</td>
</tr>
<tr>
<td>12:38 PM</td>
<td>DL</td>
<td>A</td>
<td>1:38 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
<tr>
<td>12:39 PM</td>
<td>DL</td>
<td>A</td>
<td>1:39 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:40 PM</td>
<td>DL</td>
<td>A</td>
<td>1:40 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:42 PM</td>
<td>DL</td>
<td>A</td>
<td>1:42 PM</td>
<td>L10</td>
<td>256</td>
</tr>
<tr>
<td>12:44 PM</td>
<td>DL</td>
<td>A</td>
<td>1:44 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
<tr>
<td>12:45 PM</td>
<td>DL</td>
<td>A</td>
<td>1:45 PM</td>
<td>73S</td>
<td>120</td>
</tr>
<tr>
<td>12:47 PM</td>
<td>DL</td>
<td>A</td>
<td>1:47 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:48 PM</td>
<td>DL</td>
<td>A</td>
<td>1:48 PM</td>
<td>757</td>
<td>178</td>
</tr>
<tr>
<td>12:49 PM</td>
<td>DL</td>
<td>A</td>
<td>1:49 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>11:50 AM</td>
<td>DL</td>
<td>B</td>
<td>12:50 PM</td>
<td>L10</td>
<td>256</td>
</tr>
<tr>
<td>12:04 PM</td>
<td>DL</td>
<td>B</td>
<td>1:04 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:12 PM</td>
<td>DL</td>
<td>B</td>
<td>1:12 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:18 PM</td>
<td>DL</td>
<td>B</td>
<td>1:18 PM</td>
<td>757</td>
<td>178</td>
</tr>
<tr>
<td>12:20 PM</td>
<td>DL</td>
<td>B</td>
<td>1:20 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:25 PM</td>
<td>DL</td>
<td>B</td>
<td>1:25 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:25 PM</td>
<td>DL</td>
<td>B</td>
<td>1:25 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
<tr>
<td>12:27 PM</td>
<td>DL</td>
<td>B</td>
<td>1:27 PM</td>
<td>757</td>
<td>178</td>
</tr>
<tr>
<td>12:30 PM</td>
<td>DL</td>
<td>B</td>
<td>1:30 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
<tr>
<td>12:32 PM</td>
<td>DL</td>
<td>B</td>
<td>1:32 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
<tr>
<td>12:32 PM</td>
<td>DL</td>
<td>B</td>
<td>1:32 PM</td>
<td>73S</td>
<td>120</td>
</tr>
<tr>
<td>12:34 PM</td>
<td>DL</td>
<td>B</td>
<td>1:34 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:35 PM</td>
<td>DL</td>
<td>B</td>
<td>1:35 PM</td>
<td>73S</td>
<td>120</td>
</tr>
<tr>
<td>12:36 PM</td>
<td>DL</td>
<td>B</td>
<td>1:36 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:37 PM</td>
<td>DL</td>
<td>B</td>
<td>1:37 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:37 PM</td>
<td>DL</td>
<td>B</td>
<td>1:37 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
</tbody>
</table>

a, c. The codes and abbreviations of airlines and aircraft types are referred to the Official Airline Guides.
b. The departure time of each aircraft is assumed to be 1 hour after its arrival time.

### 5. MODEL APPLICATION AND ANALYSIS
<table>
<thead>
<tr>
<th>Arrival Time</th>
<th>Airline</th>
<th>Concourse</th>
<th>Departure Time</th>
<th>Aircraft Type</th>
<th>Capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>12:40 PM</td>
<td>DL</td>
<td>B</td>
<td>1:40 PM</td>
<td>EM2</td>
<td>24</td>
</tr>
<tr>
<td>12:40 PM</td>
<td>DL</td>
<td>B</td>
<td>1:40 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:42 PM</td>
<td>DL</td>
<td>B</td>
<td>1:42 PM</td>
<td>757</td>
<td>178</td>
</tr>
<tr>
<td>12:44 PM</td>
<td>DL</td>
<td>B</td>
<td>1:44 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:45 PM</td>
<td>DL</td>
<td>B</td>
<td>1:45 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:46 PM</td>
<td>DL</td>
<td>B</td>
<td>1:46 PM</td>
<td>73S</td>
<td>120</td>
</tr>
<tr>
<td>12:47 PM</td>
<td>DL</td>
<td>B</td>
<td>1:47 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:48 PM</td>
<td>DL</td>
<td>B</td>
<td>1:48 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:49 PM</td>
<td>DL</td>
<td>B</td>
<td>1:49 PM</td>
<td>757</td>
<td>178</td>
</tr>
<tr>
<td>11:30 PM</td>
<td>J7</td>
<td>C</td>
<td>12:30 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>11:40 PM</td>
<td>J7</td>
<td>C</td>
<td>12:40 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>11:44 PM</td>
<td>TW</td>
<td>C</td>
<td>12:44 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>11:47 PM</td>
<td>TW</td>
<td>C</td>
<td>12:47 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>11:50 PM</td>
<td>J7</td>
<td>C</td>
<td>12:50 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>11:55 PM</td>
<td>J7</td>
<td>C</td>
<td>12:55 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>12:00 PM</td>
<td>J7</td>
<td>C</td>
<td>1:00 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>12:05 PM</td>
<td>KP</td>
<td>C</td>
<td>1:05 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:10 PM</td>
<td>KP</td>
<td>C</td>
<td>1:10 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:10 PM</td>
<td>J7</td>
<td>C</td>
<td>1:30 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>12:30 PM</td>
<td>J7</td>
<td>C</td>
<td>1:30 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>12:34 PM</td>
<td>TW</td>
<td>C</td>
<td>1:34 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:40 PM</td>
<td>WV</td>
<td>C</td>
<td>1:50 PM</td>
<td>73S</td>
<td>120</td>
</tr>
<tr>
<td>12:45 PM</td>
<td>J7</td>
<td>C</td>
<td>1:45 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>12:50 PM</td>
<td>J7</td>
<td>C</td>
<td>1:50 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>12:50 PM</td>
<td>TW</td>
<td>C</td>
<td>1:50 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>11:30 AM</td>
<td>CO</td>
<td>D</td>
<td>12:30 PM</td>
<td>73S</td>
<td>120</td>
</tr>
<tr>
<td>11:40 AM</td>
<td>NW</td>
<td>D</td>
<td>12:40 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>11:45 AM</td>
<td>UA</td>
<td>D</td>
<td>12:45 PM</td>
<td>73S</td>
<td>120</td>
</tr>
<tr>
<td>11:54 AM</td>
<td>UA</td>
<td>D</td>
<td>12:54 PM</td>
<td>73S</td>
<td>120</td>
</tr>
<tr>
<td>11:58 AM</td>
<td>US</td>
<td>D</td>
<td>12:58 PM</td>
<td>737</td>
<td>103</td>
</tr>
<tr>
<td>12:05 PM</td>
<td>CO</td>
<td>D</td>
<td>1:05 PM</td>
<td>735</td>
<td>108</td>
</tr>
<tr>
<td>12:11 PM</td>
<td>KP</td>
<td>D</td>
<td>1:11 PM</td>
<td>737</td>
<td>103</td>
</tr>
<tr>
<td>12:21 PM</td>
<td>US</td>
<td>D</td>
<td>1:21 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:31 PM</td>
<td>NW</td>
<td>D</td>
<td>1:31 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>12:50 PM</td>
<td>CO</td>
<td>D</td>
<td>1:50 PM</td>
<td>733</td>
<td>128</td>
</tr>
<tr>
<td>11:30 AM</td>
<td>DL</td>
<td>E</td>
<td>12:30 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>11:35 AM</td>
<td>AC</td>
<td>E</td>
<td>12:35 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
<tr>
<td>11:44 AM</td>
<td>DL</td>
<td>E</td>
<td>12:44 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>11:44 AM</td>
<td>TW</td>
<td>E</td>
<td>12:44 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>11:55 AM</td>
<td>KX</td>
<td>E</td>
<td>12:55 PM</td>
<td>737</td>
<td>103</td>
</tr>
<tr>
<td>12:05 PM</td>
<td>QF</td>
<td>E</td>
<td>1:05 PM</td>
<td>733</td>
<td>128</td>
</tr>
<tr>
<td>12:15 PM</td>
<td>LM</td>
<td>E</td>
<td>1:15 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:21 PM</td>
<td>UA</td>
<td>E</td>
<td>1:21 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:30 PM</td>
<td>DL</td>
<td>E</td>
<td>1:30 PM</td>
<td>M80</td>
<td>172</td>
</tr>
<tr>
<td>12:36 PM</td>
<td>DL</td>
<td>E</td>
<td>1:36 PM</td>
<td>757</td>
<td>178</td>
</tr>
<tr>
<td>12:50 PM</td>
<td>JM</td>
<td>E</td>
<td>1:50 PM</td>
<td>72S</td>
<td>134</td>
</tr>
<tr>
<td>12:50 PM</td>
<td>J7</td>
<td>E</td>
<td>1:50 PM</td>
<td>D9S</td>
<td>139</td>
</tr>
</tbody>
</table>

5. Model Application and Analysis
Figure 42. Travel Distance vs. Simulation Time Profile for TU No.4

Figure 43. Travel Distance vs. Simulation Time Profile for TU No.4
Traveling from Ticketing Station to Concourse A Station

5. MODEL APPLICATION AND ANALYSIS
Figure 44. Acceleration vs. Simulation Time Profile for TU No.4 Traveling from Ticketing Station to Concourse A Station

Figure 45. Acceleration vs. Distance Profile for TU No.4 Traveling from Ticketing Station to Concourse A Station

5. MODEL APPLICATION AND ANALYSIS 126
Figure 46. Velocity vs. Simulation Time Profile for TU No.4 Traveling from Ticketing Station to Concourse A Station

Figure 47. Velocity vs. Distance Profile for TU No.4 Traveling from Ticketing Station to Concourse A Station

5. MODEL APPLICATION AND ANALYSIS
Figure 48. Power Required for TU No.4 Traveling from Ticketing Station to Concourse A Station

Figure 49. Energy Consumption for TU No.4 Traveling from Ticketing Station to Concourse A Station

5. MODEL APPLICATION AND ANALYSIS
Figure 50. Energy Consumption for TU No.4

Figure 51. Load Factor for TU No.4 Traveling from Ticketing Station to Concourse A Station (Base/Horizon Year Scenario)

5. MODEL APPLICATION AND ANALYSIS
Figure 52. Occupancy for TU No.4 Traveling from Ticketing Station to Concourse A Station (Base/Horizon Year Scenario)

Figure 53. LOS in TU No.4 (Base Year Scenario)

5. MODEL APPLICATION AND ANALYSIS
Figure 54. Average Waiting Time on the Platform at North Concourse A Station (Base/Horizon Year Scenario)

Figure 55. Passenger Queuing on the Platform at North Concourse A Station (Base/Horizon Year Scenario)

5. MODEL APPLICATION AND ANALYSIS
Figure 56. LOS on the Platform at North Concourse A Station (Base/Horizon Year Scenario)

Figure 57. Number of Passengers on the Platform at North Concourse A Station (Base Year Scenario)

5. MODEL APPLICATION AND ANALYSIS
Figure 58. LOS on the Platform at North Concourse A Station (Base Year Scenario)

Figure 59. Number of Passengers Entering/Leaving the Platform at North Concourse A Station (Base Year Scenario)

5. MODEL APPLICATION AND ANALYSIS
Figure 60. LOS on the Escalator at Concourse A Station (Base/Horizon Year Scenario)

Figure 61. LOS on the Stairway at Concourse A Station (Base/Horizon Year Scenario)

5. MODEL APPLICATION AND ANALYSIS
Figure 62. Number of Passengers Arriving at Concourse A by APM (Base/Horizon Year Scenario)

Figure 63. Number of Passengers Taking the Elevator at Ticketing Station (Base/Horizon Year Scenario)

5. MODEL APPLICATION AND ANALYSIS
6. CONCLUSIONS AND RECOMMENDATIONS

6.1 Introduction

This final chapter describes some of the many issues underlying the development of the APM simulation model and the use of the EXTEND software. The recommendations are made to improve the accuracy of simulation models.

6.2 Conclusions

The APM simulation model developed through this research has been designed to provide planners and designers with information about their system. The information available from the model includes the estimation of APM system LOS in terms of waiting time and passenger space allocation in each facility, the passenger queuing, the travel time required to process a TU and passenger through the system, vehicular load factor, vehicular speed, power requirement, and energy consumption. This information is capable of assisting planners and designers in the evaluation and improvement of the performance of the APM system.

Although using computer simulation as an analysis tool in the airport APM system environment seems very natural, there is still a significant gap between conceptual model design and actual operational application to a physical system. In the APM simulation model passengers and TUs are assumed to have simple behavior. If their behavior were more complex and dynamic, the model results could be unrealistic. Some of most significant aspects of the model include the following conclusions drawn from the analysis of the model results:

(1) One of the major benefits of the APM simulation model is the ability to model the various types of APM system configurations. Together, blocks, graphics, animation
and hierarchical modeling provide the modeller with the ability to build and experiment with a model that is a visualization of objects in the modeller's application domain. For example, graphics allow the modeller to interface with the computer based on a APM network diagram, animation helps the modeller verify a model with a full understanding during simulation run, and hierarchical modeling allows the modeller to easily construct a model with reusable and pre-defined blocks.

(2) The APM simulation model is useful for estimating the effects of changes in the system by modifying the input data for the model. Therefore, simulation experiments can be conducted to test the effectiveness of modifications to the existing APM operations, such as the number of vehicles per TU, the number of TUs, cruise speed, safety headway, and station dwell time. Also, experiments can be accomplished to estimate the impact of physical changes in the system.

(3) The simulation time varies depending on the length of the simulation, number of input data, model structure, and computer type. Especially, the time required to simulate the passenger flow is much longer because passengers are simulated as individual entities. Due to the large passenger demand in the ATL APM model, it would take about 12 hours of simulation time on Power Macintosh 6100 computer with 16MB RAM.

(4) The guideway model blocks provide the capability to easily model alternative service concepts such as shuttles, loops, and single or double routes. The station model blocks enable the modeller to simulate various station systems. Three Pax-Schedule blocks help the modeller to manipulate demand matrices to achieve the specific profile desired. In addition, the Veh-Schedule block is useful to easily model alternative vehicle characteristics such as capacity, speed, and acceleration.

(5) The APM simulation Model could be applied in simulating an APM system for different scenarios including sensitivity to: passenger demand patterns, safety headways,
station dwell times, cruise speeds, breaking acceleration rates, number and types of vehicles per TU, number of TUs, and other APM operational conditions.

6.3 Recommendations

The main recommendations as a result of the experience gained in developing the APM simulation model are:

(1) To integrate the APM simulation model, an extensive library of modeling blocks catering to the various APM system requirements for different airport environments may be developed. Therefore, the model will become more flexible and friendly to use.

(2) It is necessary to increase the capability of the APM simulation model to be able to accommodate more sophisticated system control strategies, such as flexible station dwell times affected by variations in passenger demand and load/unload rates, and possibility of non-stop at some stations for particular TUs.

(3) As one part of major system at an airport the APM system had better be modeled and simulated with other relevant systems. If possible, the APM system should be associated with airport landside simulation in order to produce more accurate results.

(4) It is possible to improve the energy consumption model by considering other energy losses such as curve energy, dynamic suspension energy, and dynamic wheel-guideway energy.

(5) Analysis of an automated movers system using the simulation model developed requires sufficient data which must be accurate and representative of the systems that are being modeled.

(6) The model outputs in the APM simulation model are available only in the individual facilities except the energy consumption model. A single output or file may be designed to give a summary of the outputs of each facility.

6. Conclusions and Recommendations
(7) In order to save simulation time and memory or simplify the hierarchical block structure, some existing blocks should be modified or substituted by user-created blocks using ModL language.
BIBLIOGRAPHY

1. Airport Commissioner’s Office, Hartsfield Atlanta International Airport, Atlanta, Georgia, 1995.


*BIBLIOGRAPHY*
24. McGean, T.J., “Some Performance Factors Relevant to the Evaluation of People


28. Vandebona, U. and Richardson, A.J., “Trans: Transit Route Animation and
153-177.


*BIBLIOGRAPHY*
APPENDIX A

Generic Library Blocks Used In The Model

Add Block

This block is used for basic math purposes. It outputs the total value by adding the three or two inputs on the left of the block.

Constant Block

This block generates a constant value at each step. The constant value can be specified in the dialog and the default constant is 1. If the input is connected, the input value is added to the constant specified in dialog. This block is typically used for setting the value for the inputs to other blocks.

Conversion Table Block

This block acts like a lookup table. It contains a table of values (x in and y out) that is used to calculate an output value for a given input value. The table of values defines a curve and the output is calculated based on where the input occurs on the curve. The table can be typed into the dialog and can also be specified whether the curve is linearly interpolated or stepped. In the APM guideway model, the block is used to determine whether the TUs switch guideway or not.

Divide Block

This block used for basic math purposes divides the top input by the bottom input.
**Equation Block**

This block is used for basic math purposes and delivers error messages for checking APM system variables such as the sufficiency of dwell time for alighting passengers. This block allows to enter a free-form equation into the dialog and Extend will calculate the results. The user can apply Extend’s built-in operators and functions, and some or all of the input values as part of the equation.

**Help Block**

This block shows help text. The user can use this block to document his models. This block may also be used to include information about what the model does, what impact specific blocks or parameters have on the model, authorship, cross-references to other models, and so on. There are four text entry boxes in this block and each box can be typed up to 255 characters.

**Input Data Block**

This block generates a curve of data over time from a table of values based on time. This block is the same as the conversion table block except that the variable is always time of the simulation. In the Pax-Schedule (method 2) block of the APM station model, this block is used to generate and plot the passenger demand function with respect to time.
**Input Random Number Block**

This block generates random integers or real numbers based on the selected distribution. The user can use the dialog or the two inputs, 1 and 2, to specify arguments for the distributions. The type of distribution are: Uniform (integer or real), Binomial, Erlang, Exponential, HyperExponential, LogNormal, Normal, Poisson, Triangular, Weibull, and General. In addition, the General distribution uses a table of up to 50 values to generate a discrete, stepped, or interpolated general distribution. This block is usually used to provide a value of processing time or delay in the model.

**Multiply Block**

This block is used for basic math purposes. It multiplies the inputs.

**Select Output (5) Block**

The block passes the input value to one of five outputs according to the value of the T connector. The top output is selected if the T connector is 1 and the bottom output is selected if it is 5. For connectors that are not selected, the dialog lets the user specify an output value of either noValue, 0, or a repeat of the last value they output.

**Subtract Block**

This block is used for basic math purposes. It subtracts the bottom input from the top input.
APPENDIX B
Discrete Event Library Blocks Used In The Model

Activity, Delay Block

This block holds only one item for a specified amount of time, then releases it. The delay time is the value in the dialog or, if connected, the value at the D connector when the item is received (the connector overrides the dialog). This block can be used for any kind of service delay. For example, this block can be used to represent the delay to enter escalator, elevator and sidewalk in the APM station model.

Activity, Multiple Block

This block holds many items and passes them out based on the delay and arrival time for each item. The item with the smallest total delay and earliest arrival time is passed out first. The delay time for each item is set through the D connector or, if nothing is connected there, can be specified in the dialog. In the APM station model, this block can be used to represent a passageway facility where passengers arrive at different times and take a varying amount of time to walk through. Passengers who arrive earlier or walk faster will leave first; passengers who arrive later or walk slower will leave last.

Activity, Service Block

This block passes an item or several items only when the demand connector is connected and certain conditions exist at the demand input (either demand's value is true
[greater than 0.5] or it pulls in an item). Depending on the type of output connector (item or value) attached to demand, this block passes single item or passes a specified number of items. This block allows service on demand. This block serves as a conditional wait. In APM station model, it can be used as the transit unit arrivals, then passengers can alight and board.

**Combine Block**

This block combines the items from two different sources into a single stream of items. The items in the Combine block still retain their separate identities and are not batched together. Examples of use in the APM model are: merging traffic of transit units, passengers coming from two facilities to enter a single facility.

**Executive Block**

This block is the heart of each discrete event model and should be placed to the left of all other blocks. It controls the timing and passing of events in a discrete event model. It allows the duration of the simulation to be controlled by the end time or by another number specified in the dialog. Generally, there is no reasons to change the default values in the dialog or no other blocks to be connected to this block.
Exit Block

This block passes items out of the simulation. The total number of items absorbed by this block is reported in its dialog and at the # connector or shown on the block if animation is executed.

Gate Block

This block allows only a specified number of items to be in a restricted section of the model. It is used to restrict the passing of items into a system that can only have a specific number of items in the system at a time. The first items to arrive are allowed through, up to the number specified, and new items are allowed to pass through the block when one or more of those original items have left the restricted section. The block determines that an item has left the restricted section when the sensor connector receives an input from the output of the last block in the section. The sensor connector does not accept items, but only views them. Thus the last block in the section has its output connected in parallel: it is connected both to whichever block follows it and to the sensor input.

Generator Block

This block provides items for a discrete event simulation at specified interarrival times. Users can choose either a distribution on the left, or choose the general distribution and enter probabilities in the table. Items can be generated with a random distribution or at a constant rate of arrival. The number of items at each event can be specified in the dialog or at the V connector. The parameters for the distribution are set in the dialog. The random
distributions could include: uniform integer, uniform real, binomial, constant, Poisson, normal, log normal, exponential, Erlang, HyperExponential, Weibull, and general. The general distribution may have up to 20 points and may be interpreted as a discrete, stepped, or interpolated distribution. The input connectors 1 and 2 allow users to change the parameters of the random distribution as the simulation progresses.

**Get Attribute Block**

This block displays and/or removes attributes on items, then passes the items through. The attribute is shown in the dialog and output at the A connector. Users can use the Set Attribute block to set attributes for items. As items are passed through this block, the block reads or removes an attribute, and that attribute can be specified as the first attribute in the list or a named attribute. If the attribute is found, its value is reported in the dialog and sent through the A connector. If the attribute is not found, the user can specify whether to output a 0 or noValue at A. There is also a choice in the dialog to change the value of the item to the value of the attribute.

**Get Priority Block**

This block reads the priority of items then passes them through. The priority is shown in the dialog and output at the P connector. The priority is set using the Set Priority block. The priority number is usually used to route items. The A connector outputs 1 if the priority value has changed since the last item was read in.
**Information Block**

This block views and displays information about the items that pass through it. The first column in the table in the dialog is the time the item arrived in the block, the second column is the priority of the item, and the remaining columns are attribute values for the named attributes. If an attribute is not found, its value is blank. Because the data table is limited to 500 elements, the 501st item will write over the first item.

**Prioritizer Block**

This block prioritizes the outputs, allowing items to be sent into parallel process activities based on a user-defined priority. Items arrive through the item input connector. They are made available at one of the five item output connectors based on two things: the specified priorities, and whether or not the items are needed at the output. In order of its priority, each output will be checked to see if an item is needed. The item will leave through the highest available prioritized output. If two or more outputs have the same highest priority, the output closest to the top will be checked first to see if the item is needed; if it isn’t needed there, the next lower output with that same priority will be checked. The priorities are modified in the dialog, or can be specified through the five value input connectors, overriding the dialog values. The value input on the left corresponds to the priority for the top output connector; the value input on the right corresponds to the priority for the bottom output connector. Priorities are specified as real numbers, with the lowest number (including negative numbers) representing the highest priority.
Program Block

The block provides items by scheduling many items to be output into the model. This is similar to the Generator block, except the arrival times of the items are scheduled rather than random. The user can also assign a value, a priority, and attributes to each item generated. These items (with a given output time, item value, priority, attribute name, and attribute value) may repeat on a regular basis. This block is useful for repetitive or timed needs. Up to 500 events can be specified before repeating a sequence.

Queue, FIFO Block

This block provides a first-in-first-out (FIFO) queue. The maximum queue length, which determines how many items the queue can hold, can be set in the dialog. The user can specify that the simulation should stop when the queue is full (reaches the maximum length). The average queue length, average wait time, and utilization of the queue can be viewed in the dialog.

Queue, Matching Block

This block provides a queue in which items are released only if they have the specified attribute and the attribute's value matches the value at the ID input connector. The block searches an item entering the queue to find the attribute named in the dialog. If the attribute's value matches the value at the ID input connector, the item is released. If there is
more than one item with that attribute name and value, only the first one that entered the block is released unless the "Release single items" option in the dialog is not selected. Items which don’t have the named attribute will never be released from the queue. This block is useful for being sure that items in a simulation have a particular characteristic at a particular time. For example, this block is applied in the Guideway Maneuver block to release the passenger when the TU is released due to the same attribute value of TU number they have.

**Resource Block**

This block holds and provides items to be used in a simulation. It can be used as part of an open or closed system. This block is similar to a queue. Items can be pulled from the resource through the item output connector as long as they are available. If the block’s contents become negative, the block will not output any values until the contents become a positive number. The change connector can change the number of items stored in the resource by the value given in dialog. In the APM model, this block is only used as a FIFO queue block and the number of items stored in the resource must be set by 0.

**Select DE Input Block**

This block selects one input to be output based on a decision. The dialog has options for choosing based on the top priority of the inputs, changing which input is selected after a given number of items have passed, or choosing based on the select connector. If the select connector is not used, the user can have one out of a specified number of items come from the bottom connector or can choose based on priority. If the
select connector is used, the default is that the top input is selected with a 0 and the bottom input with a 1. The user can select whether invalid selections cause Extend to use the top connector or to wait until the connector has a valid input. The user can also specify that each true value (greater than 0.5) or item at the select connector toggles the input.

**Select DE Output Block**

This block selects the input item to be output at one of two output connectors based on a decision. The dialog has options for changing the outputs after a given number of items have passed and selecting based on the select connector. If the select connector is not used, one out of every specified number of items go to the bottom connector. If the select connector is used, the default is that the top output is selected with a 0 and the bottom output with a 1. The user can select whether invalid selections cause Extend to use the top output or to wait until the connector has a valid input. The user can also specify that each true value (greater than 0.5) or item at the select connector toggles the output.

**Set Attribute Block**

This block sets the attributes of items passing through the block. Up to seven attribute names and values could be assigned to an item with each Set Attribute block. The attributes may replace or add to existing item attributes. The user can specify the value of one of the attributes with the A connector. The value through the A connector overrides the corresponding value in the dialog.
Set Attribute (5) Block

This block works the same as the Set Attribute block except that the user uses the value input connectors to set the values for five attributes rather than just one.

Set Priority Block

This block sets a priority to items that pass through. The priority value may be assigned at the P connector or, if no connection is made there, in the dialog. The lowest value (including negative numbers) has the top priority.
APPENDIX C

Plotter and Animation Library Blocks Used in The Model

Plotter, Discrete Event Block

This plotter is used only in discrete event models. In the APM model, it is used to plot values such as information about passengers and transit units (passenger queue length, travel time, passenger waiting time, number of passengers exited, LOS values, etc.). Both the value and the event time was recorded and shown in the data table for each input. The user also can specify in the dialog whether to plot values only when they change (the default) or to plot all values.

Plotter, Scatter (4) Block

This block shows four sets of data plotted as x,y value pairs. It may be used in both continuous and discrete event simulations. The inputs are value connectors. Each pair has the two inputs of x and y. The user must connect both the x and y inputs of at least one pair in order to plot data. This block is applied in the APM Plotter hierarchical block.

Animate Item Block

This block animates, shows pictures, or plays movies when an item enters the block. The user can specify the color and pattern for the icon to animate. If this block is
used in a hierarchical window, the user can animate the hierarchical block's icon based on when items are received by the Animate Item block. First determine which block in the submodel want to be animated. Then connect the output of that block to the input of the Animate Item block, and the output of the Animate Item block to the input of the next block. The user can animate the hierarchical icon by adding an animation object to the hierarchical block's icon, then specify in the dialog of the Animate Item block what the number of the animation object is. In the APM guideway model, this block can show the transit units when passing the particular section of guideway.

Animate Value Block

This block animates, displays values, shows pictures, or plays movies in its icon based on the value at its input. The user can choose to display a value at its inputs, animate (flash box, flash text, flash circle, show picture, play movie) when the input values are greater than or equal to a value specified in the dialog. If this block is used in a hierarchical window, the user can animate the hierarchical block's icon based on the values received by the Animate Value block. First determine which block in the submodel that is animated. Then connect the output of that block to the input of the animate value block. The user animates the hierarchical icon by adding an animation object to the hierarchical block's icon. Then specify in the dialog of the Animate Value block what the number of the animation object is. In the APM model, this block can show the total number of passengers passing or entering a particular facility.
**APPENDIX D**

**Energy Consumption Model**

**The energy consumption model built in the Energy Consumption block is to calculate the TU’s travel time, speed, acceleration, power requirements, energy consumption, occupancy, load factors and LOS.**

**The Energy Consumption block is located in the Guideway Maneuver Block of each guideway block to complete an APM guideway model. Some global variables defined in the Veh-Schedule block are defined as follows:**

**Global0 = Mass of a TU**
**Global1 = Number of axles**
**Global2 = Average load per axle**
**Global3 = Cross sectional area (A)**
**Global4 = Experimental coefficient (B)**
**Global5 = Drag coefficient (C)**

**Global6 = Power efficiency**
**Global7 = Delta time for integration**
**Global15 = No. of vehicles per TU**
**Global16 = Capacity per TU**
**Global18 = Area per TU**

**The results of this model will be written into a data file defined in the Veh-Schedule model. All string variables of GlobalStr0 through GlobalStr9 are used as data files that to be open and update. Using the APM Plotter block to analyze the data and create figures.**

----------

```plaintext
real queueData[], timeArray[], nextTime;
double DataArray[][][9]; ** for data file format (9 items per row)
real lastTime, lastOutTime, currentWait;
real lenTot, waitTot;  ** arrive time total & waits total
real busyTime, lastBusy;  ** time that queue is busy
real qArea;
integer busyQFlag, timeCon, myNumber, curIndex, queueLength;
integer myIndex, itemIndex, lastLength, sending2;
integer InBlock, inConn, outBlock, outConn, connected[][];

integer max, maxSize;
integer fileNum, TUNumCon, RowNumber, fullOutNew;

real Speed, Acceleration, Energy, Power, Dist;
real Tfor, Trequired, Resistance;
real Prequired, Pconsumed, Econsumed;
real Accel, Acc1, t1, BreakDist;
real TTime, k, Tolerance, GradResist;
real LoutNew, WOutNew, maxLen;
string delim;  ** Format for data in the file

constant Output is 0;
constant Input is 1;
constant rejects is 0;
constant wants is 1;
constant taken is 2;
constant needs is 3;
constant query is 4;
```

**APPENDIX D**
constant notify is 5;
constant blocked is 6;
constant ko is 1.3;    ** Davis’ equation constant
constant kl is 29;    ** Davis’ equation constant
integer sendMsg(integer whatMsg, integer where)    ** This routine sends out a message,
                  ** and returns an answer.
{
    sysGlobalInt3 = whatMsg;
    if (where == input)
    {
        if (inBlock)    // see comment in initSim
            sendConnectorMsgToBlock(inBlock, inConn);
        else
            sendMsgToOutputs(itemIn);
    }
    else
    {
        if (outBlock)
            sendConnectorMsgToBlock(outBlock, outConn);
        else
            sendMsgToInputs(itemOut);
    }
    return(sysGlobalInt0);
}

to integer validMax()    ** This is to check the table
{
    integer i;
    I = 0;
    maxSize = 10;
    while (i<maxSize && !noValue(Force[i][0]))
    {
        i++;
    }
    if (i<=1)    ** check for at least 2 rows.
    {
        userError("Must have at least two rows of data in Speed/Tractiveforce table of block number "+MyBlockNumber());
        abort;
    }
    return(i);
}

procedure TractiveForce()
{
    integer x;

    ** This is to calculate tractive force with respect to speed.
    if (Speed < Force[0][0] || Speed > Force[max-1][0])
    {
        userError("Speed/Tractive table input (speed = "+Speed+" m/s) outside valid\n"
range in block number "+(MyBlockNumber());
  abort;
}  
else
{
  for x = 0 to (max-1)
  
    if (Speed == Force[x][0])
    {
      TFor = Force[x][1];
      break;
    }
    if (Speed < Force[x][0])
    {
      TFor = ((Speed-Force[x-1][0]))*(Force[x][1] - Force[x-1][1] - Force[x][0] - Force[x-1][0])
            + Force[x-1][1];
      break;
    }
  }  ** for loop
}  ** for else

procedure SetInitial()  ** set the initial values of Speed, Acceleration, Distance, Breaking Distance
{
  Dist   = 0.0;  ** initial distance as the beginning of this guideway
  BreakDist = 0.0;  ** Breaking distance
  LoadF   = 0.0;  ** Load factor
  t1      = 0.0;  ** t1 is used to calculate the waitDelta
  k       = currentTime;  ** k is used to create the travel time for figures
  Tolerance = 0.0000001;  ** Due to the tolerance of the numeric
  NowTime = currentTime;  ** Conversion in Modl language (see p. p. 324)

  ** This is to find and to set the begining row number of data for different TU number
  TUNumCon = TUNumIN;
  TUNum = TUNumCon;
  if( noValue(TUNumCon) || !TUNumCon)
  {
    userError("Energy Consumption block needs a TU number in block number "+(myBlockNumber()));
    abort;
  }
  if (TUNumCon == 1)  BeginRow = GlobalInt0;
  if (TUNumCon == 2)  BeginRow = GlobalInt1;
  if (TUNumCon == 3)  BeginRow = GlobalInt2;
  if (TUNumCon == 4)  BeginRow = GlobalInt3;
  if (TUNumCon == 5)  BeginRow = GlobalInt4;
  if (TUNumCon == 6)  BeginRow = GlobalInt5;
  if (TUNumCon == 7)  BeginRow = GlobalInt6;
  if (TUNumCon == 8)  BeginRow = GlobalInt7;
  if (TUNumCon == 9)  BeginRow = GlobalInt8;

APPENDIX D
if (TUNumCon == 10) BeginRow = GlobalInt9;
RowNumber = BeginRow;

Speed = SpeedIn; ** initial speed from last motion
Accel = AccelIn; ** initial acceleration from last motion
Econsumed = EnergyIn; ** initial energy from last motion
Pconsumed = PowerIn; ** initial power from last motion
TU/Direction = DirectionIn; ** Receive a TU direction Number to determine the gradient resistance
DistTra = DistIn; ** initial position of TU
TUPax = PaxIn; ** get number of pax in TU
LoadF = TUPax/Global16; ** get load factor of TU

if(noValue(Speed) || Speed < 0)
{
    userError("Energy Consumption block needs a valid initial speed value in block number "+myBlockNumber()+ ".");
    abort;
}
else // get initial speed
{
    SendMsgToAllCons(SpeedIn);
    InitSpeed = SpeedIn;
}

if(noValue(Accel))
{
    userError("Energy Consumption block needs a valid initial acceleration value in block number "+myBlockNumber()+ ".");
    abort;
}
else // get initial acceleration
    SendMsgToAllCons(AccelIn);

if(noValue(Econsumed) || Econsumed < 0)
{
    userError("Energy Consumption block needs a valid initial energy value in block number "+myBlockNumber()+ ".");
    abort;
}
else // get initial energy
    SendMsgToAllCons(EnergyIn);

if(noValue(Pconsumed) || Pconsumed < 0)
{
    userError("Energy Consumption block needs a valid initial power value in block number "+myBlockNumber()+ ".");
    abort;
}
else // get initial power
    SendMsgToAllCons(PowerIn);
if(noValue(TUDirection))
{
    userError("Energy Consumption block needs a valid direction value in block number "+myBlockNumber()+ ".");
    abort;
}
else  // get TU direction
    SendMsgToAllCons(DirectionIn);

if(noValue(DistTra))
{
    userError("Energy Consumption block needs a valid initial distance value in block number "+myBlockNumber()+ ".");
    abort;
}
else  // get initial TU position
    SendMsgToAllCons(DistIn);

if(noValue(TUPax))
{
    userError("Energy Consumption block needs a valid initial distance value in block number "+myBlockNumber()+ ".");
    abort;
}
else  // get number of Pax in TU
    SendMsgToAllCons(PaxIn);
}

procedure OpenFile()
{
    string FileNum,
    integer FileNum, BadName, Wnnten, Read, H, L;

    TUNumCon = TUNumIN;
    if( noValue(TUNumCon) || !TUNumCon)
    {
        userError("Energy Consumption block needs a TU number in block number "+(myBlockNumber()));
        abort;
    }
    if (TUNumCon == 1) FileNum = GlobalStr0;
    if (TUNumCon == 2) FileNum = GlobalStr1;
    if (TUNumCon == 3) FileNum = GlobalStr2;
    if (TUNumCon == 4) FileNum = GlobalStr3;
    if (TUNumCon == 5) FileNum = GlobalStr4;
    if (TUNumCon == 6) FileNum = GlobalStr5;
    if (TUNumCon == 7) FileNum = GlobalStr6;
    if (TUNumCon == 8) FileNum = GlobalStr7;
    if (TUNumCon == 9) FileNum = GlobalStr8;
    if (TUNumCon == !0) FileNum = GlobalStr9;
if (TUNumCon > 10) FileName = "";

badName = !FileExists(FileName) && strFind(FileName, ";", FALSE, FALSE);
if (badName)
  FileName = "";

FileNum = FileOpen(FileName, "Enter a file name for TU No. "+TUNumCon+" : ");

if (FileNum) ** Written != 0
  if (FileName == "") FileName = FileGetPathName(FileNum);

  FileClose(FileNum);       ** Close it so import and export can work

  if (BeginRow > 0)
    Read = import(FileName, "Choose a file", delim, DataArray);

  Written = export(FileName, "Enter a file name", delim, DataArray, RowNumber+1, 9);
  }
else ** Written = 0 or Read = 0
  userError("Failed to create the data file for TU No. "+TUNumCon+" in Energy Consumption block "+(MyBlockNumber())+". Please check the file name and its path.");

** The following step is to initialize an array to BLANK
  for (H=0; H< RowNumber+1; H++)
  for (L=0; L<9; L++)
    DataArray[H][L] = BLANK;

}

procedure EvsT() ** This is to create data for building the plot of energy vs travel time.
{
** create the date into a file and also change the number format (decimal places)
  MakeArray(DataArray, RowNumber+1);

  DataArray[RowNumber][0] = k;       ** k which stands for time
  DataArray[RowNumber][1] = Accel;   ** Accel which is acceleration
  DataArray[RowNumber][2] = Speed;   ** Speed which is velocity of TU
  DataArray[RowNumber][3] = DistTra; ** DistTra which stands for distance traveled
  DataArray[RowNumber][4] = Pconsumed; ** Pconsumed which stands for power consumed
  DataArray[RowNumber][5] = Econsumed; ** Econsumed which stands for energy consumed
  DataArray[RowNumber][6] = TUPax;   ** TUPax which stands for total pax in TU
  DataArray[RowNumber][7] = LoadF ;  ** LoadF which stands for load factor
  DataArray[RowNumber][8] = Global18/TUPax;  ** LOS in vehicle
  EnergyR = Econsumed;
  PowerR = PConsumed;
}

procedure Main() ** This subroutine is to calculate the velocity, distance, power, energy, acceleration.
{
  real DistToGo, Position;

APPENDIX D
SetInitial();  ** Initize all variables

Position = DistTra;  ** set initial position of TU

** This is to calculate breaking distance
if (Acc3 < 0.0)
    BreakDist = Realabs(CruiseSpeed*2/(2.0*Acc3));
else
    if (Acc3 >= 0.0)
    {
        userError("Breaking acceleration should be less than 0. The input value of"
        "+Acc3+" is invalid for this guideway in Energy Consumption block "+(myBlockNumber())+".");
        abort;
    }

BreakD = BreakDist;
if (BreakDist >= (LengthGW+EastDist+WestDist))
    {
        userError("Cruise speed is too high or breaking acceleration is too small so that the breaking
distance is greater than the length of the interstation spacing in Energy Consumption block "+(myBlockNumber())+".");
        abort;
    }

While (LengthGW - Dist > Tolerance)
    {
      EvsT();

      if (Speed <= CruiseSpeed and Accel >= 0.0)
      {
          TractiveForce();

          ** This is to calculate resistance and tractive force required.
          if (Accel != 0.0 or Speed != 0.0 or Speed != CruiseSpeed)
              Resistance = 5*10^-4*Ko+4.448*KI/Load+1.1187*10^-3*BB*Speed+239.6904*CC*AA*Speed^2/(Load*Axles);
          else
              Resistance = 5*10^-4*Ko+4.448*KI/Load+1.1187*10^-3*BB*CruiseSpeed+239.6904*CC*AA*CruiseSpeed^2/(Load*Axles);

          if (TUDirection == 1 && Gradient > 0)  **Resistance due to gradient
              GradResist = Global0*9.81^2*(Gradient/100.0);
          else
              {
                  if (TUDirection == 0 && Gradient < 0)
                      GradResist = Global0*9.81^2*(-Gradient/100.0);
                  else
                      GradResist= 0.0;
              }
          Trrequired = Resistance*Global0+GradResist;

APPENDIX D  163
This is to calculate acceleration.

if (TF0 < TRequired)
{
    userError("Tractive force required can't be greater than tractive force for acceleration in Energy\n    Consumption block "+(myBlockNumber())+". Please check the input data of TU's characteristic or
    Speed/Tractive force table.");
    abort;
}
else
    Acc1 = 1.0/Mass*(TF0-TRequired);

CurAcc = Acc1;

** This is to calculate new speed due to acceleration.
** Change DeltaT to meet the speed equaled to cruise speed and 0.

if (Accel < 0 and Speed < CruiseSpeed and BreakDist > (LengthGW+DistToGo - Dist))
{
    BreakDist = Realabs(Speed^2/(2.0*Acc3));
    BreakD = BreakDist;
}

if (TUDirection == 1) DistToGo = EastDist; ** For direction = 1 (eastbound TUs)
else DistToGo = WestDist; ** For direction = 0 (westbound TUs)

if (Speed < CruiseSpeed && (Dist < (LengthGW+DistToGo - BreakDist)))
{
    DeltaT = Global7; ** in the accelerating phase
    Accel = Acc1;
    if (((CruiseSpeed - Speed) < (Acc1*DeltaT))
        DeltaT = (CruiseSpeed-Speed)/Acc1;
    else
        if ((Dist+(Speed+Speed+Acc1*DeltaT)*DeltaT/2.0) > LengthGW)
            DeltaT = realabs((-2*Speed+sqrt((2*Speed)^2-4*Acc1*2*(Dist-
                LengthGW)))/(2*Acc1));
    else ** in the cruise phase
        if (Speed == CruiseSpeed && Dist < (LengthGW+DistToGo - BreakDist))
            DeltaT = Global7;
        Accel = 0.0;
        if ((Dist+speed*DeltaT) > (LengthGW+DistToGo - BreakDist))
            DeltaT = ((LengthGW+DistToGo - BreakDist - Dist)/Speed);
        else
            if ((Dist+speed*DeltaT) > LengthGW)
                DeltaT = ((LengthGW - Dist)/Speed);
    }
else ** in the braking phase
    if (Dist >= (LengthGW+DistToGo - BreakDist))
        DeltaT = Global7;
Accel = Acc3;
if ((Speed + Accel*DeltaT) < 0.0)
    DeltaT = Realabs(Speed/Accel);
else
    if ((Dist+(Speed+Speed+Acc3*DeltaT)*DeltaT/2.0) > LengthGW)
        DeltaT = realabs((-2*Speed+sqrt((2*Speed)^2-4*Acc3*2*(Dist-LengthGW)))/(2*Acc3));
}

** This is to calculate distance that TU is traveling.
Dist = Dist +(Speed+Speed+Accel*DeltaT)*DeltaT/2.0;
DistTra = Position+Dist;

** This is to calculate speed
Speed = Speed + Accel*DeltaT;

** This is to calculate power required.
if (Accel > 0)
    Prequired = Speed*TFor/PEff;
else
    Prequired = Speed*Trequired/PEff;

** This is to calculate power consumed.
if (Accel < 0)
    Pconsumed = 0.0;
else
    Pconsumed = Prequired; ** Power required at specific time.

** This is to calculate Energy consumed.
Econsumed = Econsumed + Pconsumed*DeltaT/(3.6*10^6); ** Accumulated energy consumed.

t1 = t1 + DeltaT; ** This is to calculate total travel time in this guideway.
k = k + DeltaT; ** This is to calculate the travel time data for EvsT files.
RowNumber ++; ** GlobalInt0 which stands for the row number
CurSpeed = Speed;
}

SpeedOut = Speed;
sendMsgToInputs(SpeedOut);
if (speed == 0 and Accel < 0) Accel = 0.0; ** Initialize the acceleration for next motion
    AccelOut = Accel;
sendMsgToInputs(AccelOut);
DistOut = DistTra;
sendMsgToInputs(DistOut);

EvsT(); ** Final read/write to array
OpenFile(); ** Write to files

if (TUNumCon == 1) GlobalInt0 = RowNumber+1;
if (TUNumCon == 2) GlobalInt1 = RowNumber+1;
if (TUNumCon == 3) GlobalInt2 = RowNumber+1;
if (TUNumCon == 4) GlobalInt3 = RowNumber+1;
if (TUNumCon == 5) GlobalInt4 = RowNumber+1;
if (TUNumCon == 6) GlobalInt5 = RowNumber+1;
if (TUNumCon == 7) GlobalInt6 = RowNumber+1;
if (TUNumCon == 8) GlobalInt7 = RowNumber+1;
if (TUNumCon == 9) GlobalInt8 = RowNumber+1;
if (TUNumCon == 10) GlobalInt9 = RowNumber+1;

EndRow = RowNumber;
EnergyOut = EConsumed;
sendMsgToInputs(EnergyOut);

if (Speed != 0 or Speed < 0.00000001)
{
    PowerOut = PConsumed;
sendMsgToInputs(PowerOut);
}

procedure setDelta()  ** This is to set delay time in this guideway according to total travel time.
{
    Main();
    waitDelta = t1;

    if (NoValue(waitDelta))  ** abort if noValue
    {
        userError("NoValue received for delay in Energy Consumption block number "+myBlockNumber()+ ".");
        abort;
    }
}

procedure utilization(integer sendMessageFlag)
{
    real currentUseFraction;

    currentUseFraction = (queueLength + (itemOut > 0.0)) / maxLen;

    ** for utilization calculation
    if( queueLength + (itemOut > 0.0) > 0 )
    {
        if( busyQFlag )
            busyTime += (currentTime - lastBusy) * currentUseFraction;

        busyQFlag = 1;
        lastBusy = currentTime;
    }
    else if( queueLength + (itemOut > 0.0) <= 0 )
        busyQFlag = 0;

APPENDIX D
if (busyTime == 0 || (currentTime - startTime) == 0)
    utilize = 0.0;
else
    utilize = busyTime/(currentTime - startTime);
}

procedure AveLength()
{
    qArea = qArea + (currentTime - lastTime) * queueLength;
    lastTime = currentTime;
    if (qArea == 0)
        lenAvg = 0.0;
    else
        lenAvg = qArea / (currentTime - startTime);
}

procedure length()
{
    LoutNew = queueLength + (itemOut > 0.0);

    if (LoutNew != lastLength)
    {
        lastLength = LoutNew;
    }

    if (queueLength + (itemOut > 0.0) >= maxLen )
    {
        if (queStop)
        {
            userError("The maximum number of TUs in this guideway is "+maxLength+" in Energy Consumption block "+(myBlockNumber())+". Please check the vehicle schedule, station dwell time, or travel time.");
            abort;
        }

        if (fullOutNew == 0)
        {
            fullOutNew = 1;  ** queue overflow
        }
    }
    else
    {
        if (fullOutNew == 1)
        {
            fullOutNew = 0;  ** queue overflow
        }
    }
}

procedure departure()

APPENDIX D 167
{ 
    aveLength();
    utilization(TRUE);

    outs += 1.0;

    currentWait += currentTime - lastOutTime;  // wait for pick up

    WoutNew = currentWait;
    waitTot += currentWait;
    waitAvg = waitTot/outs;

    length();

    if (animationOn)  // Animation
    {
        animationText(2, queueLength);
        animationShow(3);
        waitNTicks(1);
        animationHide(3, FALSE);
    }

    sendMsg(notify, output);

    // template for trace:   [BLOCK NAME ****************|block number |BLOCK NUMBER********||Current Time:|CURRENTTIME 11]
    fileWrite(sysGlobal2,"Activity,Multiple block number "+(myBlockNumber())+". Current Time:"+currentTime+"","",True);
    if(getBlockLabel(myBlockNumber()) !="")
        fileWrite(sysGlobal2,"Block Label: "+getBlockLabel(myBlockNumber(),"",True);
    fileWrite(sysGlobal2," Present Delay = "+waitDelta,"",True);
    fileWrite(sysGlobal2," Arrivals = "+Ins,"",True);
    fileWrite(sysGlobal2," Departures = "+Outs,"",True);
    fileWrite(sysGlobal2," Current number in activity = "+LoutNew,"",True);
    fileWrite(sysGlobal2," ","",True);
}

integer getItem()
{

    real  value, temp, addTime;
    integer itemGotten;

    itemGotten = FALSE;  // ** check for update queue at each event
    ** check input first so entities can be passed thru
    if ( itemIn < 0.0 && queueLength < maxLen)
        sendMsg(wants, input);

APPENDIX D
if (itemIn > 0.0) /* test for data present
{
    length();

    if( LoutNew < maxLen )
    {
        itemIndex = itemIn;
        itemIn = -itemIn;
        itemGotten = TRUE;

        setDelta();
        addTime = currentTime + waitDelta;

        EvsT();

        putRear(queueData, addTime);
        putRear(queueData, itemIndex);
        putRear(queueData, currentTime);

        aveLength();
        utilization(TRUE);
        queueLength++;
        ins++;
        length();

    if (animationOn) /* Animation
        {
            animationText(2, queueLength);
            animationShow(1);
            waitNTicks(1);
            animationHide(1, FALSE);
        } /* if this is the next item out
        ** then modify nextTime
    if( addTime < nextTime )
    {
        curIndex = queueLength - 1; /* item just put in queue
        nextTime = addTime;
    }

    if (nextTime > currentTime)
        timeArray[myIndex] = nextTime;

    sendMsg(taken, input); // item taken
    ConnectorMsgBreak();
}
return(itemGotten);
procedure getItems()
{
    integer itemGotten;
    itemGotten = TRUE;

    while (queueLength < maxLen && itemGotten)
        itemGotten = getItem();
}

integer sendItem(integer sendNeeds)
{
    real next, curItem, curNow, curTime;
    integer i, itemSent;

    itemSent = FALSE;

    ** only change output if myEvent time
    ** check output first (so ready to get input)
    if (currentTime >= nextTime )
    {
        if (itemOut <= 0.0 OR push )
        {
            next = QueLookN(queueData, curIndex*3); ** uses input time stored in queue
            if (!noValue(next) ) ** if we have something in the queue
            {
                curNow = queGetN(queueData, curIndex*3);
                curItem = queGetN(queueData, curIndex*3);
                curTime = queGetN(queueData, curIndex*3);

                utilization(TRUE);
                aveLength();
                queueLength -= 1;
                itemOut = curItem;
                itemSent = TRUE;
                currentWait = curTime - curTime;
                lastOutTime = curTime;
                if (queueLength > 0 ) ** find next closest time for output
                {
                    curIndex = 0;

                    if (timeCon )
                    {
                        for (i=i; i<queueLength; i++)
                        {
                            if (queLookN(queueData, i*3) < queLookN(queueData, curIndex*3) )
                                curIndex = i;
                        }
                    }
                    nextTime = queLookN(queueData, curIndex*3);
                }
            }
        }
    }

APPENDIX D
else
{
    curIndex = 0;
    nextTime = 1e1000;
}

if (nextTime < currentTime)
    timeArray[myIndex] = currentTime;
else
    timeArray[myIndex] = nextTime;
}
}

if (itemOut > 0.0 && sendNeeds)
    sendMsg(needs, output);  // item needs to be taken

return(itemSent);
}

procedure sendItems()
{
    integer itemSent;

    sending2 = TRUE;
    itemSent = TRUE;
    while (itemOut <= 0 && queueLength > 0 && itemSent)
    {
        if (sendMsg(wants, output) == needs)
            itemSent = sendItem(TRUE);
        else
            itemSent = FALSE;
    }
    sending2 = FALSE;
}

on itemIn
{
    integer saveArg;

    if (sysGlobalInt3 == blocked)
        return;

    if (sysGlobalInt3 == notify)
        return;

    saveArg = sysGlobalInt3;

    if (!sending2)
        sendItems();

APPENDIX D
if (saveArg == wants)
{
    if (queueLength < maxLen)
    {
        sysGlobalint0 = needs;
        connectorMsgBreak();
    }
    else
        sysGlobalint0 = rejects;
    return;
}

getItem();

if (queueLength < maxLen)
    sysGlobalint0 = needs;
else
    sysGlobalint0 = rejects;
    sysGlobalint3 = saveArg;
}

on itemOut
{
    if (sysGlobalint3 == query)
    {
        sendItem(FALSE);
        sysGlobalint0 = 0;
        if (itemOut > 0.0)
            sysGlobalint0 = itemOut;
        else if (nextTime >= currentTime && queueLength > 0)
            sysGlobalint0 = queLookN(queueData, curIndex*3+1);
        return;
    }

    if (sysGlobalint3 == taken)
    {
        departure();
        return;
    }

    if (itemOut <= 0.0 && !sending2)
        sendItem(FALSE);

    if (itemOut > 0.0)
        sysGlobalint0 = needs;
}

on Simulate ** This message occurs for each step in the simulation.
{
    if (queueLength > 0 || itemOut > 0) // this is the no simulate Msgs optimization
    {

APPENDIX D

172
sendItems();
getItems();
if (nextTime > currentTime)
    timeArray[myIndex] = nextTime;
}
utilization(TRUE);
length();
}

on checkData** If the dialog data is inconsistent for simulation, abort.
{
    holdOut = 1;
    AnimationRectangle(1);
    AnimationColor(1, 60000, 60000, 60000, 1);
    AnimationRectangle(3);
    AnimationColor(3, 60000, 60000, 60000, 1);
    AnimationText(2, "0");
    if (animationOn)
        AnimationShow(2);
    else
        AnimationHide(2, FALSE);

    if( noValue(maxLength) or (maxLength < 1))
    {
        userError("Energy Consumption block needs a valid maximum queue length in block number
"+(myBlockNumber()));
        abort;
    }

    if( noValue(LengthGW) or noValue(CruiseSpeed) or noValue(Acc3) or LengthGW < 0 or CruiseSpeed
<= 0 or Acc3 >= 0)
    {
        userError("Energy Consumption block "+(myBlockNumber())+" needs a input valid value in
dialog.");
        abort;
    }
    if( noValue(EastDist) or noValue(WestDist) or EastDist <0 or WestDist <0)
    {
        userError("Energy Consumption block "+(myBlockNumber())+" needs a input valid value in
dialog.");
        abort;
    }

    myIndex = sysGlobalint0;
sysGlobalint0 += 1.0;
    LoutNew[0]] = 10000.0; // msg to Plotter
}

on initSim ** Initialize any simulation variables.
{

APPENDIX D
myNumber = myBlockNumber();

inBlock = 0;
GetConBlocks(myNumber, 0, connected);
if (GetDimension(connected) == 1)
{
    inBlock = connected[0][0];
inConn = connected[0][1];
}

outBlock = 0;
GetConBlocks(myNumber, 1, connected);
if (GetDimension(connected) == 1)
{
    outBlock = connected[0][0];
    outConn = connected[0][1];
}
DisposeArray(connected);

queueInit(queueData); ** queue initialized
queueLength = 0;

max = validMax();
initSpeed = 0.0;
Dist = 9.0;
CurAcc = 0.0;
CurSpeed = 0.0;
DistTra = 0.0;
NowTime = CurrentTime;
Prequired = 0.0;
Pconsumed = 0.0;
Econsumed = 0.0;
Mass = Global0;
Axles = Global1;
Load = Global2;
AA = Global3;
BB = Global4;
CC = Global5;
PEff = Global6;
DeltaT = Global7;
maxLen = maxLength+1;
GradResist = 0.0;
lastBusy = startTime;
lastTime = 0.0;
lastOutTime = 0.0;
waitTot = 0.0;
WaitDelta = 0.0;
lenTot = 0.0;
qArea = 0.0;
LoutNew = 0.0;
itemOut = 0.0;
ins = 0;
outs = 0;
fullOutNew = 0;
busyTime = 0;
busyQFlag = 0;
lenAvg = 0;
waitAvg = 0;
utilize = 0;
curIndex = 0; /* next item out is the front by default
nextTime = 1e1000;
sending2 = FALSE;
delim = ",";
EnergyR = 0.0;
PowerR = 0.0;

if( getPassedArray(sysGlobal0, timeArray) )
{
    myNumber = myBlockNumber();
}
else
{
    userError("The Executive block must be present and to the left of all blocks on the worksheet");
    abort;
}
}

on endSim
{
    string str;

    utilization(FALSE); // calculate final utilization
    AveLength(); // calculate the final average number of transactions in the block

    // sysGlobal1 is the file reference number for the TEXT REPORT
    if( sysGlobal1 != 0.0 ) // 0 is error, check for open file for REPORT
    {
        // template for report: \|BLOCK NAME "***************block number \|BLOCK NUMBER**
        fileWrite(sysGlobal1,"Activity, Multiple block number "+(myBlockNumber()),\"",True);
        if(getBlockLabel(myBlockNumber()) != "")
            fileWrite(sysGlobal1,"Block Label: "+getBlockLabel(myBlockNumber()),\"",True);
        if( comments != "" )
            fileWrite(sysGlobal1," "+comments,\"",True);
        fileWrite(sysGlobal1," Input Parameters:\",\",True);
        fileWrite(sysGlobal1," Last Delay Used: "+waitDelay,\",TRUE);
        fileWrite(sysGlobal1," Maximum number allowed in activity: "+maxLength,\",TRUE);
        if(queStop)
            fileWrite(sysGlobal1," The simulation will abort if this block overflows",\",TRUE);
        if(push)
            fileWrite(sysGlobal1," This block will push items",\",TRUE);

    APPENDIX D
fileWrite(sysGlobal1," Simulation Results:",",",True);
fileWrite(sysGlobal1," Average number of transactions = "+lenAvg,"",True);
fileWrite(sysGlobal1," Average time in block = "+waitAvg,"",True);
fileWrite(sysGlobal1," Maximum number in block = "+maxLength,"",True);
fileWrite(sysGlobal1," Arrivals = "+Ins,"",True);
fileWrite(sysGlobal1," Departures = "+Outs,"",True);
fileWrite(sysGlobal1," ",",",True);

DisposeArray(queueData);

} on createBlock
{
    maxSize = 10;
    maxLength = 1000;
    queStop = 0;    ** checkbox for aborting simulation if queue full
    waitDelta = 0;
    holdOut = 1;
}
VITA

Yi-Dar Lin was born in Taipei, Taiwan, on December 18, 1967. He was graduated from the five-year program of Civil Engineering Department at National Taipei Institute of Technology, Taiwan in July 1989. After working for over one year, he came to University of Missouri-Columbia, Missouri, USA in Fall 1991 to pursue his undergraduate studies. He received his degree of Bachelor of Science in Civil Engineering in December 1993. He started pursuing his graduate studies at Virginia Polytechnic Institute and State University in Spring 1993 and then he completed his degree of Master of Science in Civil Engineering, specializing in Transportation Engineering in September 1995. He plans to study Ph.D. program to continue his higher education.