LOCATION PROBLEMS IN THE PRESENCE OF QUEUEING

by

SAMUEL SHIN-WAI CHIU
B. A., BEREA COLLEGE (1974)

SUBMITTED TO THE SLOAN SCHOOL OF MANAGEMENT IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 1982

Samuel Shin-Wai Chiu 1982

The author hereby grants to M.I.T. permission to reproduce and to distribute copies of this thesis document in whole or in part.

Signature of Author.......................................................... Sloan School of Management December 16, 1981

Certified 1, .......................................................... Richard C. Larson
Supervisor

Accepted by.......................................................... Jeremy F. Shapiro
Chairman, Interdepartmental Graduate Committee

Archives

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

APR 05 1982

LIBRARIES
LOCATION PROBLEMS IN THE PRESENCE OF QUEUEING

by

Samuel Shin-Wai Chiu

Submitted to the Sloan School of Management on December 16, 1981 in partial fulfillment of the requirements for the Degree of Doctor of Philosophy.

ABSTRACT

Determining the optimal location of facilities to serve a spatially dispersed set of customers has been an increasingly active research area during the past twenty years. Many systems whose facilities are to be located operate in a stochastic environment, with uncertainties associated with the arrival time, location and service requirements of each customer. Facilities having finite service capacities can experience queueing-like congestion in such environments. Unfortunately, only a small fraction of locational research has included stochastic phenomena in the problem formulation.

In this thesis, our goal is to begin to build an integrated theory linking traditional location theory to the theory of stochastic processes, particularly queueing processes, to arrive at improved methods for locating facilities operating in a stochastic environment. We assume that each facility garages one or more mobile servers which travel to the scene of customer service requests. The locational objective is the minimization of average system response time to random customer, where response time is the sum of travel time (the standard measure of performance in traditional location models) and queueing delay; in applications this second term may be considerably larger than travel time.

We first formulate and solve a stochastic loss problem in which a single facility is to be located to station n mobile servers. In this system, a call for service is "lost" (i.e. to be served by a back-up unit) at a cost if, upon arrival, all servers are busy answering previous customer demands. The optimal location is one which minimizes the weighted cost of "loss" and response time (in this case, the travel time). This optimal location is shown to coincide with a minisum site -- a location that minimizes the average travel time. This equivalence is true regardless of topological setting and distance metric measure as long as demands for service are generated by a time-homogeneous Poisson process. The cost of "loss" is, curiously enough, only constrained to be non-negative.

We next consider a system where calls for service enter an infinite capacity queue to await service (on a first-come-first-serve basis) if all servers are busy. We focus our attention on the optimal location of one single mobile server on a tree network. Customer arrival pattern follows a time-homogeneous Poisson process generated solely on the nodes
of the network. Convexity properties of the average response time on a tree allow us to develop an efficient procedure for finding the optimal location, which we will call the Stochastic Queue Median (SQM). Non-linearity of the objective function introduced by the presence of the second moment of service time, and singularity caused by possible system saturation make the analysis non-trivial. We also analytically trace the trajectory of the SQM as total traffic intensity (arrival rate) is varied.

A natural generalization of the above SQM problem is to allow demands to be continuously distributed on links of the network as well as discretely on the nodes. We obtain parallel results in this generalization, again on a tree network. A by-product of this investigation, which is of interest in its own right, provides new information regarding minisum location on a general undirected network with continuous link and discrete nodal demands. Numerical examples are constructed to illustrate concepts and algorithmic procedures throughout this thesis.

Finally, to initiate further research efforts on locational decisions in the presence of queueing, we formulate and indicate possible solution strategies for several potentially important problems.

THESIS SUPERVISOR: Richard C. Larson
TITLE: Professor of Electrical Engineering and Urban Studies.
DEDICATED TO MY PARENTS
ACKNOWLEDGEMENTS

I wish to express my deepest appreciation to Professor Richard C. Larson for his guidance and encouragement during my thesis research. His enthusiasm for and dedication to research is contagious and "contaminating". Dick's quest for excellence is inspirational and it shows on the tennis court. I also want to thank Liz for her genuine hospitality.

I thank Professors Stephen C. Graves, Thomas L. Magnanti and Amedeo R. Odoni for serving on my thesis committee. Their careful reading and comments of my thesis are greatly appreciated.

Al and Liz Drake have been constant sources of encouragement during my entire graduate study at M.I.T.. Their genuine concerns for my personal as well as professional development are very much appreciated.

I truly enjoyed my stay at M.I.T. and appreciate deeply the friendship of Paulo and Lena, Horacio and Corina, Vajaya and Uma, Sarkar, Singhal, Rick, Paul, Jim and Rich. I have already started to miss terribly the time we spent on and off the tennis courts.

Also, recognition is due to the M.I.T. Operations Research Center for providing a 'home' for my graduate career. I also want to thank the Center for Advanced Engineering Study for providing me with a very valuable experience in teaching an introductory Operations Research course.

Constant encouragement and understanding from my family keeps me going in the toughest of times. My parents deserve more credit than words can convey. My sisters Alice and Mimi, my brother Simon and my brother-in-law Kai form the closest knit of family one can ask for. I cannot leave out Sung Gae and Sung Wae in my acknowledgement because they are the best nephews one can have.

I believe Kathy Sumera's name appears in more theses than anyone else's because of her reliable and excellent technical typing.

Before getting carried away and turning this acknowledgement into a book, I would like to thank AW for giving me time, and myself for finishing my degree.
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>2</td>
</tr>
<tr>
<td>Acknowledgements</td>
<td>5</td>
</tr>
<tr>
<td>Table of Contents</td>
<td>6</td>
</tr>
<tr>
<td>List of Tables</td>
<td>8</td>
</tr>
<tr>
<td>List of Figures</td>
<td>9</td>
</tr>
<tr>
<td>Chapter 1: Introduction</td>
<td>11</td>
</tr>
<tr>
<td>1.1 Thesis Objective</td>
<td>11</td>
</tr>
<tr>
<td>1.2 Overview</td>
<td>13</td>
</tr>
<tr>
<td>1.3 Problem Formulations of the Stochastic Loss and Stochastic Queue Systems</td>
<td>18</td>
</tr>
<tr>
<td>1.4 Organization of the Thesis</td>
<td>22</td>
</tr>
<tr>
<td>Chapter 2: On Categorizing Location Problems with Queueing-Like Congestions</td>
<td>24</td>
</tr>
<tr>
<td>2.1 Complexity of Queueing-Location Problems</td>
<td>24</td>
</tr>
<tr>
<td>2.2 System Components of a Queueing-Location Problem</td>
<td>26</td>
</tr>
<tr>
<td>Chapter 3: Stochastic Loss System</td>
<td>35</td>
</tr>
<tr>
<td>3.1 Introduction</td>
<td>35</td>
</tr>
<tr>
<td>3.2 One Server Stochastic Loss System</td>
<td>36</td>
</tr>
<tr>
<td>3.3 The n-Server Single-Facility Stochastic Loss System</td>
<td>44</td>
</tr>
<tr>
<td>3.4 Stochastic Loss System Under General Topological Settings and Demand Distributions</td>
<td>51</td>
</tr>
<tr>
<td>3.5 Discussion</td>
<td>52</td>
</tr>
<tr>
<td>Chapter 4: Stochastic Queue Median on a Tree Network</td>
<td>Page</td>
</tr>
<tr>
<td>--------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>4.1 Introduction</td>
<td>56</td>
</tr>
<tr>
<td>4.2 Stochastic Queue Median on a Tree Network</td>
<td>56</td>
</tr>
<tr>
<td>4.3 Parametric Analysis (on ( \lambda )) of the SQM Location</td>
<td>58</td>
</tr>
<tr>
<td>4.4 The Optimal Value Function in ( \lambda )</td>
<td>85</td>
</tr>
<tr>
<td>4.5 Postscript</td>
<td>104</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 5: Stochastic Queue Median on a Tree Network with Continuous Link Demands</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1 Preliminary</td>
<td>123</td>
</tr>
<tr>
<td>5.2 The Minisum Location on a Network with Continuous Link Demands</td>
<td>123</td>
</tr>
<tr>
<td>5.3 Properties of ( s^2 ) and ( TR ) on a Tree Network</td>
<td>125</td>
</tr>
<tr>
<td>5.4 Locating the SQM and Parametric Analysis on a Tree Network with Continuous Link Demands</td>
<td>146</td>
</tr>
<tr>
<td>5.5 A Numerical Example</td>
<td>159</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 6: Formulations of Several Queueing-Locatiion Problems</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Preliminary</td>
<td>169</td>
</tr>
<tr>
<td>6.2 Problem Formulations</td>
<td>169</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 7: Conclusions and Summary of Results</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1 Summary of Results</td>
<td>170</td>
</tr>
<tr>
<td>7.2 Concluding Remarks</td>
<td>180</td>
</tr>
</tbody>
</table>

References | 182 |

| References | 184 |
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.5.1</td>
<td>Values of $s^2$ for Selected Nodes</td>
<td>163</td>
</tr>
<tr>
<td>5.5.2</td>
<td>Values of $s^{2'}$, $\bar{s}$, $\bar{t}'$</td>
<td>166</td>
</tr>
</tbody>
</table>
List of Figures

<table>
<thead>
<tr>
<th>Figure Number</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3.1</td>
<td>Temporal Sequence of a Service Demand</td>
<td>20</td>
</tr>
<tr>
<td>3.2.1</td>
<td>Breakpoint $x_B$ on Link $(a,b)$</td>
<td>40</td>
</tr>
<tr>
<td>3.2.2</td>
<td>Possible Interior Minimum of $Z(x)$</td>
<td>42</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Temporal Sequence of a Service Demand</td>
<td>59</td>
</tr>
<tr>
<td>4.2.2</td>
<td>A Tree Rooted at Node $i$</td>
<td>74</td>
</tr>
<tr>
<td>4.2.3</td>
<td>A Step in the Trimming Algorithm</td>
<td>77</td>
</tr>
<tr>
<td>4.2.4</td>
<td>Sketch of $G(x)$</td>
<td>81</td>
</tr>
<tr>
<td>4.2.5</td>
<td>A Binary Tree with Four Levels</td>
<td>84</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Sketch of $A(x,\lambda)$ in</td>
<td>89</td>
</tr>
<tr>
<td>4.3.2</td>
<td>DTX Profiles at $x_1$ and $x_2$</td>
<td>92</td>
</tr>
<tr>
<td>4.3.3</td>
<td>$A(x,\lambda)$ at Two Points on the Same Link</td>
<td>95</td>
</tr>
<tr>
<td>4.3.4</td>
<td>$A(x,\lambda)$ at Node $j$</td>
<td>95</td>
</tr>
<tr>
<td>4.3.5</td>
<td>DTX Profiles and Trajectory of SQM</td>
<td>98</td>
</tr>
<tr>
<td>4.3.6</td>
<td>DTX Profiles and Trajectory of SQM</td>
<td>100</td>
</tr>
<tr>
<td>4.3.7</td>
<td>The MSP Rooted at Node $1$</td>
<td>101</td>
</tr>
<tr>
<td>4.3.8</td>
<td>DTX Profiles with Multiple Minimum Locations</td>
<td>101</td>
</tr>
<tr>
<td>4.3.9</td>
<td>A Numerical Example to Find $\lambda^-$ and $\lambda^+$</td>
<td>103</td>
</tr>
<tr>
<td>4.4.1 a</td>
<td>Range of $\lambda$ and DTX Profiles</td>
<td>106</td>
</tr>
<tr>
<td>4.4.1 b</td>
<td>Range of $\lambda$ and DTX Profiles</td>
<td>107</td>
</tr>
<tr>
<td>4.4.2</td>
<td>$\theta(x,\lambda)$ and $A(x,\lambda)$ for Fixed Value of $x$</td>
<td>113</td>
</tr>
<tr>
<td>4.5.1</td>
<td>Multiple Local Optima of $TR(x)$ on a Link</td>
<td>116</td>
</tr>
<tr>
<td>5.2.1</td>
<td>No ABL Links Associated with Facilities Located on $(a,b)$</td>
<td>133</td>
</tr>
<tr>
<td>5.2.2</td>
<td>Segmentation of Link $(i,j)$, an ABL Link</td>
<td>133</td>
</tr>
<tr>
<td>Section</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>5.2.3</td>
<td>Segmentation of Link ((i,j)), an AL Link</td>
<td>133</td>
</tr>
<tr>
<td>5.2.4</td>
<td>Minisum Location at Node (j)</td>
<td>141</td>
</tr>
<tr>
<td>5.2.5</td>
<td>Graph of (F_L(x)) and (y)</td>
<td>141</td>
</tr>
<tr>
<td>5.2.6</td>
<td>Flow Chart: Minisum Location on a Tree Network with Continuous Link Demands</td>
<td>143</td>
</tr>
<tr>
<td>5.2.7</td>
<td>An Example: Minisum Location on a Tree Network with Continuous Link Demands</td>
<td>145</td>
</tr>
<tr>
<td>5.5.1</td>
<td>Numerical Example: SQM on a Tree Network with Continuous Link Demands</td>
<td>162</td>
</tr>
<tr>
<td>5.5.2</td>
<td>MSP Associated with Figure 5.5.1</td>
<td>165</td>
</tr>
<tr>
<td>5.5.3</td>
<td>DTX Profiles of Numerical Example in Figure 5.5.1</td>
<td>167</td>
</tr>
</tbody>
</table>
Chapter 1
INTRODUCTION

1.1 Thesis Objective

The motivation of this study comes from consideration of facility location problems where spatially dispersed customers receive service from mobile and/or stationary servers. Our primary concern is with locational decisions in an urban environment comprised of a local transportation network and spatially dispersed system users. In the public domain, examples include police patrol deployments, ambulance services, fire departments and public emergency repair services. In the private sector, examples include repair services for utilities and other service-oriented products (e.g., dishwasher repairs, window and lock replacements and computer equipment servicings), dial-a-ride systems and taxi-cab fleets. In other instances where customers travel to a facility to receive services, we have banking services, super-markets, and department stores.

In many classical locational decisions, minimization of average system-wide travel time has been the sole objective. This decision criterion implies that servers are available at all times and each call for service is answered immediately by the nearest (always available) server. Under such a congestion free environment, average travel time is a good measure of system performance. In an urban environment, service-oriented systems are plagued with uncertainties: spatial uncertainties as to the location of the next call for service; temporal uncertainties as to the arrival time of the next customer; service time uncertainties as to the amount of time required to complete a service; travel time uncertainties as to the transit time fluctuations on an urban transportation network.
For example, in an urban emergency service setting where system utilization (i.e., when all servers are busy) is high, queueing delay is often an order of magnitude larger than the travel time. Only a year ago in New York City, it was all too common to wait 45 minutes before an ambulance was available to answer a call for service. Thus, one naturally attempts to define a performance measure which includes both queueing delay and travel time -- the average system-wide response time. Ignoring queueing delay in the objective function can lead to locational decisions far from optimal.

The main objective of this thesis is to merge the concerns of queueing theory and location theory. We want to explicitly incorporate the stochastic nature of queueing delays into locational decisions. The initial attempt is to apply known queueing results to facility location problems on a network. Queueing-location problems are rich in structure due to the uncertainties inherited in a spatial (probability distribution of demands for service) and temporal (probabilistic arrival time and service time for each customer) setting. New service discipline rules (which dictate the operating policy of a queueing system) may emerge as a result of research efforts being expanded in this area. We hope that future research activities in this direction will open up new avenues in queueing theory. When this happens, the above-mentioned integration will be complete in both directions: from queueing theory to locational decisions and from locational considerations to queueing systems.

Relative lack of research effort in this area does not reflect any value judgements about the importance of stochastic problems in a locational context. Tansel, Francis and Lowe [72] suggest that this negligence of probabilistic consideration is probably due to the bifurcation
of the profession into "optimizers" and "probabilizers". It is with this realization that we direct our research effort in this thesis.

In integrating the concerns of location theory and queueing theory, the potential number of problems can be explosively large. Thus, one must develop a systematic procedure to select problems for analysis. We identify certain important system elements which are central to a queueing-location problem (in Chapter 2), but stop short of introducing a classification scheme. This is because we feel that more planning and additional insights are required before one can devise a meaningful categorization.

To demonstrate solution methodology and problem tractability, we formulate and analyze several queueing-location problems. The method of analysis is novel and we hope it will provide insights into problem structures and solution procedures along this direction of research.

We will review the relevant literature on locational decisions in the next section. Section 1.3 introduces two queueing-location models from Berman, Larson and Chiu [3], which will initiate the analysis of Chapters 3, 4 and 5. We conclude this chapter with a directory establishing a road map for the rest of the thesis.

1.2 Overview

This section contains a brief literature survey relevant to queueing-location problems. A very recent survey paper by Tansel, Francis and Lowe [72] provides an excellent literature review which includes state-of-the-art information concerning locational decisions on a network. We will first focus on location theory literature dating back to 1964 with the seminal work of Hakimi ([26], [27]). One will notice the lack of coordinated effort in merging the concerns of queueing and locational
theory as we move on to review the relevant literature on locational decisions in a stochastic environment.

The problem of determining the locations of a set of \( p \) facilities on a network that minimizes the expected travel time to and/or from the facilities, for the user population on the nodes of the network, is one of the classic problems in location theory. This problem, known in the literature as the \( p \)-median problem, has been studied extensively in the last two decades. The basic theoretical results are due to Hakimi ([26], [27]), who shows that one only has to consider nodal locations as candidates. Hua [34] and Goldman [22] independently devise efficient algorithms to locate a single median on a tree network. Mirchandani and Oudjit [60] develop theorems and algorithms to limit the search of the two-median problem on a tree network.

In the past twenty years, researchers have studied many variations of the \( p \)-median problem. Levy [51] considers capacity constraints on facilities, while Hakimi and Maheshwari [28], Wendell and Hurter [78] impose upper limits on link capacities. Goldman [21] generalizes the median problem to consider multi-commodities (as identified by origin-destination pairs) going through two-stage processing at the nearest facilities. Subsequently, Hakimi and Maheshwari [28], Wendell and Hurter [78] extend the analysis to the case of multistage processing. Slater [69] partitions the nodes of the network, and seeks a location such that the sum of the distances from the facility to a nearest element of each component (of the partitioned node set) is minimized. Levy [51] proves nodal optimality when the linear distance objective function is replaced by concave cost functions of the distance between each node and its nearest median. Goldman [23] specializes this to the one-median
problem on a tree network when the objective function is defined by the sum of polynomial functions of distance from any node to the median.

In an entirely different topological setting, Larson and Li [48], Larson and Sadiq [50], Odoni and Sadiq [65], analyze routing and location problems on a plane with an inhomogeneous travel medium (i.e., with impenetrable barriers or high speed freeways). While our classification concern does capture such geometrical settings, we will focus our attention on a network topology.

From an algorithmic perspective, procedures have been developed to locate a set of nodal facilities. The focus is on the minimization of total travel time. Mirchandani [58] generalizes the nodal optimality result to the optimization of a convex utility function. Search procedures (see Handler and Mirchandani [31]) include: enumeration methods (Hakimi [26]); graphic-theoretic approaches (Goldman [21], Matula and Kolde [56], Kariv and Hakimi [37]); heuristic procedures (Maranzana [54], Kuehn and Hamberger [43], Cornuejols, Fisher and Nemhauser [14], Cooper [12], Teitz and Bart [74], Järvinen, Rajada and Sinervo [35]); primal-based mathematical programming algorithms (Eilon, Watson-Gandy and Christofides [17], Efroymson and Ray [16], Khumorwala [40], Revelle and Swain [66], Schrage [67]); partitioning algorithms (Geoffrion and Graves [20], Magnanti and Wong [53]); dual-based mathematical programming (Cornuejols et al [14], Bitran, Chandru, Sempolinski and Shapiro [6], Erlenkotter [18]); and the cross optimization approach (Van Roy [76]). In many of the applications, a fixed charge is levied for each facility location and the number of facilities to be located becomes a decision variable. The objective function becomes the cost of service (e.g., travel time), fixed set-up cost and the cost of facility maintenance.
To study location problems in a stochastic environment, one must consider at least four elements of uncertainties: (i) spatial uncertainties as to the locations of demands; (ii) temporal uncertainties, as to the arrival times of customers; (iii) service time uncertainties as to the amount of time required to complete a given service; and (iv) network state uncertainties as to the probabilistic behavior of link travel time. One must incorporate the above uncertainties in any realistic locational model operating as a spatially dispersed system in a stochastic setting.

A number of probabilistic versions of the p-median problem have been considered. They capture a subset of the above mentioned elements of uncertainties, but usually assume away the contribution of queueing delay to system response time, or they ignore the dependence of system state probabilities (a state represents server(s) status - idle or busy) on the location of the facility. Frank [19] considers the one-median problem when the nodal weights are random variables and link lengths are deterministic. Mirchandani and Odoni [59] consider the case where network link lengths are random variables and nodal weights and known constants. Berman, Larson and Odoni [4] classify the ongoing median-related research effort into four categories: (P1) medians on stochastic networks; (P2) movable servers on stochastic networks; (P3) congested medians; and (P4) congested movable servers systems. Without loss of optimality, the search for facilities can be limited to nodes of the network in (P1) and (P2). In a stochastic loss system (where demands for service are generated independently in a Poisson manner at each node, customer requests are served by back-up units at a cost when the lone server is busy), we again obtain the familiar nodal optimality result (Berman, Larson and Chiu [3]). Stochastic analyses of the median problem have centered on the changes of
network travel times at discrete time epochs. In the case of movable server systems, researchers use Markovian decision theory to devise re-location (of facilities) policies (Berman [1], Berman and Odoni [5]).

Previous consideration of queueing as a component of travel time relies on the assumption of exponential service time distribution, which allows us to model the system as a Markov process. Mirchandani, Silve and Visocki [6] study the resultant Markov model for a network with two nodes. Berman and Larson [2] generalize this and show that at least one set of optimal locations exist on the nodes of a network.

Larson's Hypercube Queueing Model [44] provides a way to evaluate several system performance measures for alternative facilities locations. He generates steady-state equations of the underlying Markov process under different operating policies. He then solves for the steady-state probabilities of the system and hence the set of expected system performance measures for a given set of locations. Larson's approach is descriptive, in the sense that his algorithm computes a set of performance measures (average travel time, server workloads, inter-district dispatching frequency, etc.), and the choice of "optimal location" is left to the planner in an interactive mode (by relocating facilities and recomputing performance measures) with the computer. In the case when average travel time is the single objective, the nodal optimality result of Berman and Larson [2] enables the search to be limited to the nodes of the network.

In all of the above location problems, demands are generated solely on the nodes of the network. In the actuality of an urban emergency setting, accidents do occur everywhere on the network (e.g., highways -- links). Handler and Mirchandani [31] formulate the p-median problem on
a general network with discrete nodal and continuous link demands. On a
tree network, they propose a solution procedure, which is a slight
variant of the Goldman algorithm [22] for the one-median problem.
Minieka [57] implicitly considers link demands in a surrogate way. Speci-
fically, he defines a general absolute median of a network to be a point
on the network that minimizes the (unweighted) sum of the distances from
that point to the most distance point on each link. No systematic
analysis has been performed to study the behavior of the expected travel
distance (as a function of facility location) on a general (or even a
tree) network.

Section 1.3 describes two stochastic location models of Berman,
Larson and Chiu [3]. The last section of this chapter provides a
directory for the organization of the thesis.

1.3 Problem Formulations of the Stochastic Loss and Stochastic Queue
Systems

In this section, we review briefly the model of Berman, Larson and
Chiu [3], which will serve as a starting point of our thesis.

Problem Structure:

Let $G(N,L)$ be an undirected network with node set $N(|N| = n)$ and
link set $L$. Service demands occur exclusively at the nodes, with each
node $i$ generating an independent time-homogeneous Poisson stream of rate
\[ \lambda h_i \] (\( \sum h_i = 1 \)). Travel distance on link $(i,j)$ is $d_{ij}$. Travel time is
equal to travel distance divided by a constant speed $v$. The time required
to travel a fraction $\theta$ of link $(i,j)$ is $\theta \frac{d_{ij}}{v}$. All travel distances
are taken as the shortest distance between two points on the network,
denoted by $d(x,y)$ for $x, y$ on $G(N,L)$. 
Model Assumptions:

- Demands for service arise solely on the nodes of $G$ in a time-homogeneous Poisson manner.
- A single mobile server resides at a facility located at $x$ on link $(a,b)$ of $G$.
- The server, when available, is dispatched immediately to any demand that occurs.

Loss Model:

When an arriving customer finds the server busy with a previous demand, it is rejected at a cost $Q$, $Q \geq 0$.

Stochastic Model:

Unanswered demands enter into a queue that is depleted in a first-come, first-server (FCFS) manner upon the availability of the server.

Objective:

It is desired to locate a facility on $G$ such that the average cost of response is minimized. For the Loss model, it is a weighted sum of mean travel time and the cost of rejection. In the Queue model, it corresponds to the sum of mean queueing delay and mean travel time.

Both models share the following common parameters. For a facility located at $x$ on $G$, the total service time associated with a random service demand is

$$\tilde{s}(x) = \tilde{w}_j + \beta/v \ d(x,j) \quad \text{with probability } h_j$$

Where $\tilde{w}_j$ is the non-travel component of service time, a random variable, $\beta/v \ d(x,j)$ accounts for round-trip travel time between facility
service demand occurs from node $j$ arrival of server at node $j$ return to facility arrival at facility end of service

d($x,j)/v \quad (3-1)d(x,j)/v \quad \text{Time}

\text{travel time to the scene} \quad \text{on-scene service time} \quad \text{travel time back off-scene to facility service time}

\text{Total service time } S(x) \text{ with probability } h_j

Figure 1.3.1 Temporal Sequence of a Service Demand
at x and incident at node j. \( \beta \) is a constant bigger than 1. Figure 1.3.1 shows the temporal sequence associated with a service demand originated from node j.

Where \( \tilde{w}_j \) = (on-scene + off-scene) service time. We also define

\[
\bar{w} = \sum_{j \in N} \bar{w}_j,
\]

where (-) indicates the expectation of a random variable; and

\[
\bar{w}^2 = \sum_{j \in N} \bar{w}_j^2
\]

\( s^2(x) \) = second moment of service time. We assume \( \bar{w}_j, \bar{w}_j^2 \) are finite for all \( j \in N \).

**Loss Model:**

The expected cost of travel to a random demand is

\[
Z(x) = [1 - P(x)] \sum_{j \in A} h_{j} d(x, j) / v + P(x)Q \sum_{j \in A} h_{j} d(x, j)
\]

for a facility located at x on G

where Q is the cost of rejection when the server is busy and P(x) is the average fraction of time that the server is busy, given that the facility is located at x on G.

One would like to find a point \( y \) on G, such that \( Z(y) \leq Z(x) \) for all \( x \in G \). We will call such a point the stochastic loss one-median. Berman, Larson and Chiu show that without loss of optimality, one can search over the nodes of the network only.

**Queue Model:**

For a facility located at x on G, the expected response time to a random request is

\[
\bar{R}(x) = \begin{cases} 
\frac{\lambda s^2(x)}{2(1-\lambda s(x))} + \bar{t}(x) & \text{when } 1-\lambda s(x) > 0 \\
\infty & \text{otherwise}
\end{cases}
\]
where
\[ \frac{\lambda s^2(x)}{2(1-\lambda s(x))} \] is the queueing delay

\[ t(x) \] is the travel time (= \( \Sigma \frac{h \cdot d(x,j)}{v} \)), and

\[ s(x) = w + \beta t(x) \]

The objective is to find a point \( y \) on \( G \), such that \( TR(y) \leq TR(x) \) for all \( x \in G \). We will term the point \( y \) the Stochastic Queue-One Median, or simply SQM. There is no nodal optimality result for this problem.

In the next section, we will lay down a road map for the remainder of this thesis.

1.4 Organization of the Thesis

This thesis is organized into seven chapters. Chapter 1 contains the usual literature review of relevant previous research efforts and the formulation of two stochastic location models. In Chapter 2, we discuss issues and essential system elements of a queueing location model when one attempts to categorize such problems. We stop short of providing a classification scheme because much more work must be done to carefully and meaningfully characterize each problem.

Chapter 3 focuses on the analysis of the stochastic loss one-median problem (i.e., zero queue capacity) in which a facility is to be located to house \( n \) servers. In Chapter 4, we analyze the stochastic queue one-median system on a tree network in its entirety. Starting with the convexity of the response time on the network, we develop efficient search procedures to optimally locate the facility. We conclude Chapter 4 with a parametric analysis of the trajectory of the optimal location when total
network traffic intensity, $\lambda$, is varied. Chapter 5 generalizes the results of Chapter 4 to a case where continuous link demands are allowed on the network. It begins with an analysis of the minisum location problem (with continuous link demands) and ends with a numerical example illustrating the search procedure and provides results of the parametric analysis.

Chapter 6 contains formulations of five analytically interesting and potentially useful queueing-location models. Each of them captures some of the important system elements discussed in Chapter 2. We also provide conjectures regarding solution techniques and optimal solution structures after the formulation of each model. The last chapter, Chapter 7, summarizes the results of each chapter. The potential for future research is emphasized throughout Chapters 1, 2 and 6.
Chapter 2
ON CATEGORIZING LOCATION PROBLEMS WITH
QUEUEING-LIKE CONGESTIONS

Our original intention was to introduce a classification scheme under which each instance of queueing-location problems can be identified by a coded vector representation. However, we feel that one needs more planning and insight before a permanent and well-structured categorization rule can be formulated. Therefore, we choose to delay such an endeavor at the present time. Instead, we will discuss features of a stochastic system central to locational decisions in this chapter. We will begin the chapter with a general discussion of the complexity inherited in a queueing-location problem. In Section 2.2, we will consider individual components of a location problem in a stochastic environment, some of which are queueing specific, while others are related to the spatial structure of location problems. It is our hope that the discussion in this chapter will lay the foundation for, and eventually lead to, the construction of a permanent classification scheme.

2.1 Complexity of Queueing-Location Problems

As mentioned in Chapter 1, when one deals with locational decisions for a spatially dispersed system operating in a stochastic environment, there are at least four basic kinds of uncertainties: spatial distribution of demand patterns; inter-arrival time between successive calls for service; service time distribution and transit time fluctuations. The potential number of research problems can be very large. Thus, it is desirable to develop a systematic way to select and identify problems for analysis.
Our main concern is to incorporate queueing delays into locational decisions. Travel time uncertainties are usually modelled by discrete changes of transit times on any given network links at discrete epochs. This modelling approach naturally fits into a Markovian decision theory framework. At each change of the network state (i.e., network transit time), a decision is made to reposition any (or all) mobile servers in anticipation of serving future demands. The objective, then, is to derive a set of optimal relocation (of servers) strategies under each network state such that the average long term cost (measured in terms of response time) to the system is minimized. It is our intention not to consider such complications at the present time. We hope that the merge of queueing-location and Markovian-relocation problems will be the focus of research in the not-so-distant future. Therefore, the travel time, within the scope of our current interest, will be determined solely by the topological structure under which our location problem is defined, and will not be probabilistic in nature.

Limiting our scope to deterministic transit times, we still have to deal with the spatial and temporal uncertainties discussed earlier. An added level of uncertainty concerns the degree of urgency which one associates with each call for service. The performance of a queueing system depends on the manner upon which waiting customers are selected to be served by the next available server. There are many service rules, unique to spatially dispersed environments, which dictate the operating policy of a queueing system. We will discuss them in more detail in Section 2.2. Other queueing specific considerations include: queue capacity; the number of servers (or facilities) to be located and the manner under which they share (or divide) their workloads -- a districting
problem; and the decision to serve specific customers by back-up units. Locational specific issues include: the topological structure under which our location problem is defined (e.g., on a network, on a plane with Euclidean or rectilinear metric); and the spatial distribution of demands for service. While our primary concern is the minimization of response time, we may wish to optimize a response-time induced utility function to account for the risk preference of our decision maker.

The above brief discussion reveals the many components and complexities abounding in a queueing-location problem. In the next section, we will examine each system component in greater detail. We hope that our discussion here will eventually lead to the birth of a classification scheme under which each instance of queueing-location problems can be uniquely identified and categorized.

2.2 System Components of a Queueing-Location Problem

2.2.1 Common System Characteristics:

The systems we are interested in share the following common characteristics:

(i) Customers arrive at spatially dispersed locations, in a time-homogeneous Poisson fashion.

(ii) In response to each call for service, either a mobile unit located at a facility is assigned to the customer, or the customer travels to a near-by service facility. In both cases, queueing-like congestion may delay the servicing of customers.

(iii) The system operates in statistical equilibrium.
All travel times are deterministic and governed by system topological structure.

The basic system performance measure is average response time which is comprised of queueing delay and travel time.

2.2.2 **System Components of a Queueing Location Problem:**

In addition to the common characteristics mentioned above, we will examine each of the essential system elements (highlighted in Section 2.1) in more detail. They are:

A. the form of the objective function;

B. the topological structure under which our locational decision is defined;

C. queue capacity;

D. number of facilities (or servers) to be located;

E. the service discipline which dictates the operating policy of our queueing system;

F. a pruning procedure which designates certain class(es) of customers to be served by back-up units.

We now proceed to examine the issues involved with each of the system components.

A. **Objective Function:**

The basic system performance measure is average response time. One may induce a utility (or loss) function on the response time and seek to optimize the expected utility when making locational decisions. Since in a life-and-death situation, which is all too common in an urban emergency setting, extreme values of response time may be intolerable, a
utility function will explicitly incorporate risk preference. In a stochastic setting, constraint optimization (such as minimizing the average system-wide response time subject to a maximum response time constraint) may not be appropriate because there is always the possibility of extreme delay in responding to a call for service. Therefore, the concept of utility optimization seems appealing.

Other objectives include: (i) minimizing the probability that the sum of weighted distances exceed a certain tolerance limit([Frank [19]]); (ii) multiobjective optimization. Halpern [29] and Handler [30] study the cent-dian (center-median bi-objective) problem, Lowe [52], Tansel, Francis and Lowe [73] study multiobjective vector minimization and minimax problems, Larson's Hypercube Queueing model [44] considers several system performance measures (such as workload balancing, average travel time, etc.).

Other forms of objective functions, which are of interest to us, include bounds and approximations of response time in more complex queueing formulations. We do not wish to exclude potentially interesting and practical problem instances where closed form expressions for average queueing delay do not exist in current literature. Such approximations give upper and lower bounds on the mean queueing delay (see, for example, Brumelle [8], Kollerstrom [42], Boxma et. al. [7] and Hokstad [32]), examination of which can be insightful in the absence of exact closed form solutions.

There is yet another solution methodology we do not wish to ignore; it is the use of simulation to study system behavior and, ultimately, the use of it to aid making locational decisions.
B. Topological Structure:

While our main concern is in topological structure with a network setting, one may wish to study location problems on a plane. Under this system element, we also specify the spatial manner in which demand pattern is distributed. In particular, we will consider the following settings:

(i) Network Structure: A network is defined on a graph \( G(N,L) \) (with node set \( N \) and link set \( L \)) together with a distance matrix which specifies the distance between every pair of nodes in \( N \). The network under study can be directed or undirected. It may have special structure, such as a tree or a complete network. The distance matrix may or may not obey the triangular inequalities. Customers may originate solely from nodes of the network or the arrival process can be distributed probabilistically along the links.

(ii) Plane Structure: Carter, Chaiken and Ignall [9] design response areas for two emergency units on a plane. More recently, Larson and Li [48], Larson and Sadiq [50], Sadiq and Odoni [65] study routing and location decisions on a plane with an inhomogeneous travel medium and discrete demands. They use \( \ell_1 \) metric as a measure of distance. One may have different measures of distance and continuous spatial distribution of arrival processes on the plane. Larson and Odoni [49] provide an excellent tutorial on geometric probability which can aid the analysis of this class of problems.
C. Queue Capacity:

Queue capacity may range from zero to infinity. In the zero capacity case, all arriving customers who find no idle server will be served (or lost) by back-up units at a specified cost. When queue capacity is non-zero but finite, customers are served by back-up units, again at a cost, when all "waiting spaces" are taken. One may consider queue capacity a decision variable when making locational decisions. We assume all customers are eventually served, by primary or back-up units.

D. Number of Servers and Facilities:

The number of facilities to be located can range from one to infinity. Each facility, in turn, can house one or more servers. Most closed form expressions for average system behavior exist only in the case of one single server. The primary research effort should focus on the one server system. However, there exist bounds and approximations results when there are many servers operating simultaneously to provide service to the same pool of customers. Also, one may partition the pool of customers into independent districts to be served by different facilities. This districting issue is the topic of discussion next, under the system element of service discipline.

E. Service Discipline:

Performance of a system operating with congestions in a spatially dispersed environment, depends critically on the manner upon which customers are selected for service by the next available server. It also depends on the coordinated assignment of servers to customers when there are more than one facility (or server) in the system. Other than the usual selection rules (selecting waiting customers for service) of first-
come-first-served (FCFS), first-come-last-served (FCLS) and service-in-
random-order (SIRO), we identify the following operating policies relevant
to a spatially dispersed system:

(i) **Priority Class Discipline**: In an urban emergency setting,
certain types of calls for service evidently require more
urgent attention than others. A popular and important
discipline categorizes customers into $K$ ($1 \leq K < \infty$)
priority classes. The proposed service discipline offers
FCFS service within a priority class, and always prefers
customers in a higher priority class. Within this operat-
ing policy, one may incorporate pre-emption or the lack
thereof. Pre-emption relates to the interruption of a
service upon arrival of a higher priority customer. A
pre-emptive policy can be of two types: pre-emptive
resume, in which service is restored at the point of
interruption; or pre-emption repeat, in which case the
entire service must be repeated (perhaps re-sampling from
the service time distribution).

(ii) **Preference Probability**: Each spatial call for service
may possess a preference probability vector to select
the more preferred available server. Larson's Hypercube
Queueing model [44] utilizes such preference structure
in making location (of servers) and allocation (of cus-
tomers) decisions.

(iii) **Spatially-Oriented Priority Discipline**: We identify
four types of service disciplines relevant and unique
to the spatially distributed system.
(a) **Minimal expected service time discipline:**
Under this policy, the next available server is assigned to the closest-in-waiting customer (or the customer with minimum expected service time if non-travel related service time differs among customers). This service discipline has the property of reducing the mean system response time when compared to a FCFS discipline.

(b) **Maximum dispatch radius:**
In which customers located more than a critical distance away from the facility are lost at a cost, whereas others are entered in queue.

(c) **Priority-conditioned dispatch radius:**
Which is the same as above, except that the critical rejection radius is dependent on customer priority.

(d) **Maximum marginal loss reduction:**
In which a nonlinear loss function is used to characterize system performance and the next customer is selected on the basis of minimizing expected future loss. This service scheme has the flavor of sequential decision making. Markovian decision theory may be required to analyze such problems.

(iv) **Districting Consideration:** When one wants to locate more than one facility, there is a districting issue one should consider. It is the manner in which the system operates: whether as one single unit or many independent systems.
They are:

(a) Each facility operates independently with no sharing of work loads among facilities. The main concern in this case is to characterize the response area for each of the facilities. We may identify such a discipline as independent operation policy.

(b) All servers (facilities) operate jointly with complete or partial sharing of work loads. One can view this as a "pooled" system where all customers are "pooled" together and served under one single operating system. Dispatching rules as to the assignment of idle servers to customers are to be specified under the service discipline described above. The exact analysis of such a system is intrinsically difficult due to the non-Markovian nature of the service time distribution. However, bounds and approximations of average response time can be used to study system behavior — usually at extreme values (high or low) of traffic intensity (i.e., arrival rate λ).

F. Pruning:

The idea here is to designate certain customers (spatially) as loss customers (to prune away) to be served by back-up units at a cost, regardless of server(s) status (i.e., idle or busy). There are several variations within this pruning procedure:

(i) **Spatially Selective Pruning**: A selected subset of spatial customers (nodal, link on a network setting and region on
a plane) are designated as loss customers. The remaining customers will be served by the primary service facility(s) to be located.

(ii) **Uniform Pruning**: All spatial customers are pruned uniformly. The net effect is to reduce the overall total arrival rate.

(iii) **Stationary Pruning**: Spatial customers are pruned away (perhaps by a mixture of selective and uniform pruning) so that the original facility location (before pruning) remains unchanged. This operation is desirable when there is a change of spatial distribution of customers after the location of a facility.

(iv) **Priority-Oriented Pruning**: Instead of spatially-pruned away customers, one may serve certain priority class customers by back-up units. A facility will be located to serve the remaining customers.

We hope that our discussion of system elements, relevant to a queueing-location problem, will contribute to the understanding and selection of future research activities in this area. It is our belief that a classification scheme will soon emerge as a result of more research efforts expanded in this direction in the not-so-distant future. In the next three chapters, we will formulate and analyze several instances of queueing-location problems, each of which derives its system descriptors from the issues raised in this chapter. Chapter 6 contains the formulations of five potentially tractable location problems with possible real-world application. There is a summary of results (by chapter) in Chapter 7.
Chapter 3
STOCHASTIC LOSS SYSTEM

3.1 Introduction

We mentioned in Chapter 1 that most of the previous research efforts have focused on the assumption that the server(s) are available at all times, and have thus ignored the stochastic nature of queueing delay. Infinite server capacities imply that the closest (or most preferred) facility is always available to serve a particular demand point. This assumption is inappropriate when the demand rate is high. When the closest (or most preferred) facility is unavailable at the time of a particular demand, the demand could be served by the closest (or most preferred) available facility, or the demand could wait in a queue until the preferred unit has a free server, or the demand could be lost at a cost, or something else could happen depending on the operating policies of the system. Previous queueing considerations in this context relied on the assumption of exponential service time distribution, which allows us to model the system as a Markov process. Mirchandani, Silve and Visocki [61] study the resultant Markov model for a network with two nodes. Berman and Larson [2] extended the nodal result to the case where availability of servers is a random variable. They assume that the state probabilities of server status remain fixed for local changes of service facility locations (except when the change of facility locations result in changes of the server preference list at demand nodes). This approximation is good when on-scene service time is much larger than travel time. They consider three different cases when all servers are busy: (i) demand is lost at a cost; (ii) demand enters a first-in, first-out infinite capa-
city queue; upon completion of service, the server is either assigned to the next request waiting in queue, or returns home immediately if none is waiting; (iii) service time is exponentially distributed. They show that the expected travel time is linear on each link and thus the nodal-optimality result. Berman, Larson and Chiu [3] prove the nodal result for the one server loss case (demand is lost at a cost when the lone server is busy). In this chapter, we extend the nodal result when there is more than one server to be located at the same facility. The result here is both more general and more restricted than the problem studied in Berman and Larson [2]. It is more general because we make no assumptions on service time distribution and we allow the steady state probabilities to vary continuously as we change facility location. It is more restrictive because we are locating one single home for all the servers.

We will first introduce and study the one-server loss system in Section 3.2, to be followed by the formulation and analysis of the n-server-single-facility loss system in Section 3.3. In Section 3.4, we show that the standard average travel time minimizing location coincides with the n-server-single-facility median in any topological setting with any demand distribution over the region of interest. The situations include continuous link demands on a general undirected network, minisum location problems on a Euclidian plane, and location problems on a plane with $l_1$ (or $l_p$) metric. The chapter ends with a discussion of the difficulties one encounters in generalizing the stochastic loss model.

3.2 One Server Stochastic Loss System

3.2.1 Model Formulation:

We want to locate a single facility for one mobile server on a net-
work. Calls for service arrive independently on each node as Poisson Processes. When the server is busy, any arriving call is lost at a cost \( Q > 0 \). The objective is to minimize the expected response time (weighted by the travel time when the server is available and by \( Q \) when the server is busy).

As formulated in Chapter 1, the objective function is

\[
Z(x) = (1 - P_1(x)) \sum_{j \in N} h_j d(j,x)/v + Q P_1(x)
\]

for a facility located at \( x \in G(N,L) \)

where

\[
P_1(x) = \text{probability that the server is busy when the facility is located at } x \text{ in } G,
\]

\[
d(j,x) = \text{shortest distance from node } j \text{ to point } x \text{ as defined by network topology},
\]

\[
h_j = \text{fraction of calls originated from node } j,
\]

\[
v = \text{travel speed}.
\]

**Definition:** \( y \in G(N,L) \) is the one-server-stochastic-loss median (1-SSL median) if \( Z(y) \leq Z(x) \) for all \( x \in G \).

For completeness, we will derive the algebraic expression of \( \sum_{j \in N} h_j d(j,x) \).

Consider a point \( x \) on link \((a,b)\) of length \( \ell \). When we say \( x \) on \((a,b)\), we mean that the point \( x \) is on link \((a,b)\) at a distance \( x \) from node \( a \).

Then

\[
d(j,x) = \min \{ d(a,j) + x, (\ell-x) + d(b,j) \}
\]
That is, the shortest path from $j$ to $x$ will pass through either node $a$ or node $b$. We partition the node set $N$ into $A(a,b;x)$ and $B(a,b;x)$, or simply $A$, $B$ if the context leaves no ambiguity, as follows:

$$A = \{j | j \in N, \text{ and } d(a,j) + x < (\lambda - x) + d(b,j)\}$$

$$B = N - A$$

We observe that the sets $A$ and $B$ may change as we move from node $a$ to node $b$ on a general undirected network.

**Definition:** Primary region of a link is the portion of a link over which the sets $A$ and $B$ are unchanged. Note that there may be as many as $|N| - 2$ primary regions on a link.

**Definition:** A break point on a link is a point on the link where there is a change in the sets $A$ and $B$.

With the above partition of the node set $N$, we can write

$$\bar{t}(x) = \frac{1}{v} \sum_{j \in N} h_j d(j,x) = \frac{1}{v} (c_1 x + c_2)$$

where

$$c_1 = \sum_{j \in A} h_j - \sum_{j \in B} h_j$$

$$c_2 = \sum_{j \in A} h_j d(a,j) + \sum_{j \in B} h_j (\lambda + d(b,j))$$

It is easy to evaluate $P_1(x)$ when we view $P_1(x)$ as the fraction of time the server is busy. If $s(x)$ is the expected service time and $\lambda$ is the total Poisson demand rate, $P_1(x)$ is:
This is because the expected length of the idle period is $1/\lambda$ for Poisson arrivals and the expected length of the busy period is $\bar{s}(x)$ for a system with zero queue capacity. $\bar{s}(x)$ can be written as

$$\bar{s}(x) = \bar{w} + \beta \bar{t}(x)$$

where $\bar{w}$ is the non-travel component of service time and $\beta \bar{t}(x)$ represents round-trip travel time ($\beta > 1$). Having fully developed the algebraic expression of $Z(x)$, we are now in the position of investigating the behavior of $Z(x)$ as $x$ moves from one end of the link to the other.

3.2.2 Nodal Result for the 1-SSL Median Problem:

We first study the behavior of $Z(x)$ over a primary region where the constants $c_1$ and $c_2$ remain fixed. We will then examine the directional derivative of $Z(x)$ in and out of a break point.

Let $$\rho(x) = \lambda \bar{s}(x) = \lambda \bar{w} + \lambda \beta \bar{t}(x)$$

Lemma 3.2.1: $Z(x)$ is monotone over a primary region.

Proof: $$Z(x) = \frac{1}{1 + \rho(x)} [\bar{t}(x) + \rho(x)Q]$$

where $$\rho(x) = \lambda \bar{s}(x) = \lambda \bar{w} + \lambda \beta \bar{t}(x)$$

and $$\bar{t}(x) = \frac{1}{v} (c_1 x + c_2)$$
\[
\frac{dZ(x)}{dx} = \frac{c_1 1 + \lambda \delta \varphi + \omega}{\nu [1 + \rho(x)]^2}
\]

\(\frac{dZ(x)}{dx}\) takes on the sign of \(c_1\) in a primary region. \(Z(x)\) is monotone in a primary region.

Note that \(Z(x)\) is not necessarily concave in a primary region. We now investigate the change of \(c_1\) as one moves across a break point \(x_B\) on link \((a,b)\) at a distance \(x_B\) from node a (refer to Figure 3.2.1).

Let \(X_1\) and \(X_2\) be two consecutive primary regions, where

\[x_1 \leq x_2 \text{ for all } x_1 \in X_1, \ x_2 \in X_2\]

and

\[X_1 \cap X_2 = x_B\]

Let \(C_1', C_2', A', B'\) and \(C_1'', C_2'', A'', B''\) denote the constant and node partitions (as defined in 3.1) for \(X_1\) and \(X_2\) respectively. Since \(X_B\) is a break point, this implies

\[\delta(x_B) = \{j | j \in N \text{ and } d(j,a) + x_B = d(j,b) + x_B \} \neq \emptyset\]
We can easily show that

\[ A' = A'' \cup \delta(x_B) \]

\[ B'' = B' \cup \delta(x_B) \]

\[ \therefore C'' = \sum_{j \in A''} h_j - \sum_{j \in B''} h_j \]

\[ = C' - 2 \sum_{j \in \delta(x_B)} h_j \leq C' \]

The above observation allows us to establish that

\[ \frac{dZ(x_B)}{dx^-} \geq \frac{dZ(x_B)}{dx^+} \]

where \( \frac{dZ(x_B)}{dx^-} \) and \( \frac{dZ(x_B)}{dx^+} \) denote the left (approaching from) derivative and the right (moving away) derivative at \( x_B \) respectively. This is true due to the fact that \( (x) \) is continuous at \( x_B \) and \( C_1 \) is non-increasing across \( x_B \).

Because \( Z(x) \) may not be concave in a primary region, we do not have a concave function over the entire link and thus the Hakimi type of nodal optimality argument does not apply here. However, the monotonicity of \( Z(x) \) over a primary region and the fact that \( \frac{dZ(x_B)}{dx^-} \geq \frac{dZ(x_B)}{dx^+} \) exclude the possibility of an interior point (of a link) being a local minimum of \( Z(x) \). The only possible interior local minimum of \( Z(x) \) is shown in Figure 3.2.2. The dotted lines in this figure denote the boundary of the primary regions. However, (a) is not allowed because \( C_1 \) decreases across the break points and (b) is not allowed because \( Z(x) \) is monotone
Figure 3.2.2 Possible Interior Minimum of $Z(x)$
in a primary region. Therefore we conclude that:

Theorem 3.2.1: The 1-SSL median exists on nodes of a network. Berman Larson and Chiu [3] prove an even stronger result.

Theorem 3.2.2: The 1-SSL median coincides with the Hakimi median [3].

The Hakimi median is a point \( x \) on \( G \), which minimizes \( \bar{t}(x) \). The nodal result of Hakimi ([26], [27]) guarantees the nodal location of the 1-SSL median.

It is surprising to note that the value of \( Q \) (cost to the system when a call is lost), except for the non-negativity requirement \( (Q \geq 0) \), does not enter the analysis of our problem. This is due to some cancellation when we differentiate \( Z(x) \) with respect to \( x \). If we impose the additional requirement that \( Q \geq t(x) \) for all \( x \in G \) (i.e., \( Q \) is an upper limit of response time, so that the system is being penalized when the server is busy), we can prove concavity of \( Z(x) \) on a link, which is accomplished next.

Lemma 3.2.2: \( Z(x) \) is concave on a link if \( Q \geq \bar{t}(x) \) for all \( x \in G \).

Proof: Because \( C_1 \) is non-increasing across the break point, we need only to prove that \( Z(x) \) is concave in a primary region.

\[
Z(x) = (1 - P_1(x)) \bar{t}(x) + P_1(x)Q = \bar{t}(x) + P_1(x)(Q - \bar{t}(x))
\]

\[
\frac{d^2Z(x)}{dx^2} = (Q - t(x)) \frac{d^2P_1(x)}{dx^2} - 2 \frac{C_1}{v} \frac{dP_1(x)}{dx}
\]

\[
= (Q - t(x)) \left( \frac{\lambda \beta}{v} C_1 \right)^2 \left( \frac{-2}{(1 + \rho(x))^3} \right) - 2 \left( \frac{C_1}{v} \right) \left( \frac{\lambda \beta C_1}{v} \right) \frac{1}{(1 + \rho(x))^2} \leq 0.
\]
"\[ Z(x) \] is concave over a primary region."

The nodal result for the 1-SSL median follows immediately from Lemma 3.2.2.

3.3 The N-Server Single-Facility Stochastic Loss System

In this section, we consider the optimal location of an n-server stochastic loss system. We want to locate a single facility for n mobile servers on a network. Calls for service arrive independently on each node as Poisson Processes. Calls originating from each node have a preference probability for each idle server as defined by system status. System status is represented by an n-tuple of the zero-one vector, \( y \). The \( i \)th component, \( y_i \), is zero if the \( i \)th server is free and one when it is busy. We denote the collection of system status vectors by \( Y \). There are \( 2^n \) elements in \( Y \). We also denote the saturation state (where all servers are busy) by \( S = (1, 1, \ldots, 1) \). In state \( S \), calls will be lost at a cost \( Q > 0 \) to the system. The objective is to minimize expected costs (measured in terms of response time) to the system.

3.3.1 Model Specification

Other than the usual network descriptions, \( G(N,A) \), call rate \( \lambda h_j \), we have the following additional problem structure.

- Calls from node \( j \) have a server preference probability vector \( \alpha_j^y = (\alpha_{jk}^y) \), where \( \alpha_{jk}^y \) = probability that a node-\( j \) call chooses server \( k \) when the system status is \( y \).

Clearly \( \alpha_{jk}^y = 0 \) if \( y_k = 1 \), i.e., when server \( k \) is busy, and \( \sum_k \alpha_{jk}^y = 1 \). When one component \( \alpha_{jk}^y = 1 \) for a particular \( k \), the server preference probability vector becomes a server preference list.
P_y = probability that the system is in state y.

In state S = (1, 1,...,1), all arriving calls are lost at a cost Q.

We wish to find an x ∈ G(N,A), such that the expected response time is minimized. We call such an optimal x the n-server-single-facility-loss median (n-SSFL median).

Before we analyze and search for the n-SSFL median, we want to make a few observations:

- For a facility located at x, the service time distribution depends only on call location and not on system status; nor does it depend on server identity.
- We are locating one single facility for all n servers.
- The location of the facility determines not only the service time distribution for each node – it also determines the steady state probabilities. Previous works on median problems with congestion (Berman and Larson [2]) assume that the steady state probabilities are given and not directly affected by the perturbation of server locations, or that the service time is exponentially distributed.
- For a facility located at x on G, the objective function takes on the following form:

\[
Z_n(x) = \sum_{y \in Y} \sum_{j \in N} P_{y}(x) \frac{h_j(x,j)}{v} + P_s(x)Q \\
= (1 - P_s(x))\bar{\tau}(x) + P_s(x)Q
\]

where \(\bar{\tau}(x) = \sum_{j \in N} h_j(x,j) / v\)
The objective function depends only on the saturation probability $P_s$, response time $\bar{t}(x)$ and penalty, $Q$.

- All servers are indistinguishable in terms of their service time distribution and hence the expected service time. We need only know the expression for $P_s$. This is readily available as the Erlang Loss Formula (Takács [71]):

$$
P_s = \frac{\rho^n/n!}{\sum_{i=0}^{n} \frac{\rho^i}{i!}}
$$

where we have suppressed the argument $x$, and

$$
\rho = \lambda w + \lambda \beta \bar{t}(x) = \lambda \bar{s}(x).
$$

$$
\bar{t}(x) = \frac{1}{v} (c_1 x + c_2)
$$

- We will first impose the condition that $Q > \bar{t}(x)$ for all $x$ on $G$ and prove a nodal result for the $n$-SSFL median.

3.3.2 Nodal Result for the $n$-SSFL Median:

**Lemma 3.3.1:** $Z_n(x)$ is monotone over a primary region if $Q > \bar{t}(x)$ for all $x$ on $G$.

**Proof:** We will evaluate $\frac{dZ_n(x)}{dx}$. Again we suppress argument $x$. 
\[
\frac{dZ_n(x)}{dx} = (1 - P_s) \frac{c_1}{v} + (Q - \bar{c}) \lambda \beta c_1 \frac{dP_s}{d\rho}
\]

where \( \rho = \lambda \bar{w} + \lambda \beta / v \) \( (c_1 x + c_2) \)

\[
\frac{dZ_n}{dx} = \frac{c_1}{v} \left[ (1 - P_s) + (Q - \bar{c}) \lambda \beta \frac{dP_s}{d\rho} \right] \quad \text{note } 1 - P_s > 0
\]

therefore \( \frac{dZ_n}{dx} \) takes on the sign of \( c_1 \) if we can show that

\[
\frac{dP_s}{d\rho} \geq 0 \text{ for all } x \text{ on a primary region.}
\]

Using

\[
P_s = \frac{\rho^n / n!}{\sum \rho^i / i!} \quad i=0
\]

we now evaluate

\[
\frac{dP_s}{d\rho}
\]

\[
\frac{dP_s}{d\rho} = \left[ \sum \rho^i / i! \right]^{-2} \frac{\rho^{n-1}}{(n-1)!} \left[ \sum \rho^i \frac{n-i}{i!n} \right] \geq 0.
\]

Therefore \( Z_n(x) \) is monotone and takes on the sign of \( c_1 \) over a primary region.

Again, since \( c_1 \) is non-increasing across break points, the same argument of Theorem 3.1.1 leads to:

**Theorem 3.3.1:** Without loss of optimality, the n-SSFL median exists on a node of a network if \( Q \geq \bar{c}(x), \forall x \in G \).
As a by-product of Lemma 3.3.1, we can prove that the n-SSFL median coincides with the Standard Hakimi median (minimizing expected travel time $\bar{t}(x)$).

**Theorem 3.3.2:** The n-SSFL median coincides with the Hakimi median if $Q \geq \bar{t}(x) \forall x \in G$.

**Proof:** Since the Hakimi median minimizes $\bar{t}(x)$, we need only show that

$$\frac{dZ_n(x)}{dt(x)} \geq 0 \text{ for all } x \in G.$$ 

$$\frac{dZ_n(x)}{dt(x)} = (1 - P_s) + \lambda \beta (Q - \bar{t}) \frac{dP}{dP} \geq 0$$

since $1 - P_s > 0$, $Q - \bar{t} > 0$ and $\frac{dP}{dP} \geq 0$ from lemma 3.3.1.

When we examine the one server loss median problem, we observe that, due to cancellation, the condition $Q \geq \bar{t}(x)$ for all $x \in G$ is not necessary to prove the nodal (or Hakimi) result. As we can see from analysis of the n-SSFL median problem, the condition $Q \geq \bar{t}$ is convenient in proving the nodal result. We need only prove the non-negativity of $\frac{dP_s}{dP}$. We would like to relax this condition and see if the cancellation effect observed in the 1-SSL median problem carries over to the generalized problem. The answer to this is yes.

**Lemma 3.3.2:** $Z_n(x)$ is monotone over a primary region.

**Proof:** From the proof of lemma 3.3.1,

$$\frac{dZ_n(x)}{dx} = \frac{c}{v} \left[ (1 - P_s) - \lambda \beta t \frac{dP}{dP} + \lambda \beta Q \frac{dP}{dP} \right]$$
We need to show that the expression in brackets is non-negative. In particular, to show

\[(1 - P_s) - \lambda \beta t \frac{dP}{d\rho} \geq 0.\]

Note:

\[(1 - P_s) - \lambda \beta t \frac{dP}{d\rho} \geq (1 - P_s) - \rho \frac{dP}{d\rho} \]

because \(\rho = \lambda \overline{w} + \lambda \beta t \geq \lambda \beta t \) and \(\frac{dP}{d\rho} \geq 0.\)

\[(1 - P_s) - \rho \frac{dP}{d\rho} =\]

\[= \sum_{i=0}^{n} \frac{\rho^i}{i!} \left( \frac{\rho^{n-1}}{(n-1)!} \right) \sum_{i=0}^{n} \frac{\rho^i}{i!} - \rho \frac{\rho^{n-1}}{(n-1)!} \sum_{i=0}^{n} \frac{\rho^i}{i!} - \rho^n/n! \sum_{i=0}^{n} \frac{\rho^i}{i!} \]

\[= A + B\]

After some algebraic manipulation, B is seen to be

\[B = \sum_{k=0}^{n-1} \frac{\rho^{n+k}}{n-k} \frac{n!k!}{n-k} \]

We need only concern ourselves with terms in A that involve \(\rho^{n+k}\) for \(k = 0, 1, \ldots, n-1\). We compute the coefficients of \(\rho^{n+k}\) in A to be

\[C(\rho^{n+k}) = \sum_{j=k+1}^{n} \frac{1}{j!(n-j+k)!} \]

for \(k = 0, 1, 2, \ldots, n-1\)

\[= \frac{1}{(k+1)!(n-1)!} + \frac{1}{(k+2)!(n-2)!} + \frac{1}{(k+3)!(n-3)!} + \ldots + \frac{1}{n!k!} \]

\((n-k) \text{ terms}\)
We see also that \( n!k! \geq (n-i)!(k+i)! \) for \( i \leq n-k \) iff

\[
n(n-1)\ldots(n-i+1) \geq (k+i)(k+i-1)\ldots(k+1)
\]

Compare the above inequality term by term.

\[
(n-l) \text{ vs } (k+i-l)
\]

\[
l = 0, 1, 2, \ldots, i = 1 \quad \text{and we know } i \leq n-k.
\]

It is clear that

\[
(n-l) - (k+i-l)
\]

\[
= n-k-i \geq (n-k) - (n-k) = 0 \quad \text{since } i \leq n-k.
\]

Therefore, \( n!k! \geq (n-i)!(k+i)! \)

or the coefficient of \( \rho^{n+k} \) (in A) is:

\[
C(\rho^{n+k}) \geq (n-k) \left( \frac{1}{n!k!} \right)
\]

\[
\therefore A - B \geq \sum_{k=0}^{n-1} \rho^{n+k} \left[ \frac{n-k}{n!k!} - \frac{n-k}{n!k!} \right] = 0.
\]

thus,

\[
(1 - P_s) - \lambda \beta \frac{dP}{dp} s \geq 0,
\]

\[\frac{dZ(x)}{dx}\] takes on the sign of \( C_1 \) in a primary region. Therefore, \( Z_n(x) \)

is monotone in a primary region. ■

An argument similar to Theorem 3.3.1 leads to nodal results of the

n-SSFL median without the assumption of \( Q \geq \tau(x) \forall x \in G \). Exactly the

same reasoning gives us the equivalence between the Hakimi median and the

n-SSFL median:
Theorem 3.3.3: The \( n \)-SSFL median coincides with the Hakimi median

Proof: From the proof of Lemma 3.3.2

\[
\frac{dZ_n(x)}{dt(x)} = (1 - P_s) - \lambda \beta \frac{dP}{dp} + \lambda \beta Q \frac{dP}{dp} \geq 0
\]

Since the Hakimi median minimizes \( \bar{t}(x) \), it also minimizes \( Z_n(x) \).

3.4 Stochastic Loss System under General Topological Settings and Demand Distributions

Consider an \( n \)-server-single-facility loss system as described above, with the exception that we are operating under a general topological setting and any demand (for service) pattern. The topological setting can be on an Euclidean plane, or any \( \lambda \) metric. The demand pattern is completely general if on a plane. On a general undirected network, we may have general continuous link demands. (We will study the stochastic queue one-median problem on a general network with continuous link demands in Chapter 5.) We will define a minisum location as a point, in the region of interest (network or plane), which minimizes the average travel time \( \bar{t} \). \( \bar{t} \) is computed under the relevant metric measure on a topological space.

The \( n \)-SSFL median is defined, as before, as the location which minimizes the weighted sum of travel time (when not all servers are busy) and the cost of rejection, \( Q \), (when all servers are busy). The objective function to be minimized in the \( n \)-server-single-facility loss system is \( Z_n(x) \), where \( x \) is the location of the facility:

\[
Z_n(x) = [1 - P_s(x)] \bar{t}(x) + P_s(x)Q
\]
where $P_s(x) =$ probability of saturation; i.e., all servers are busy, and

$$
P_s(x) = \frac{[\lambda s(x)]^n / n!}{\sum_{i=0}^{n} \frac{[\lambda s(x)]^i}{i!}} ,
$$

the Erlang loss formula, and

$$
s(x) = w + \beta t(x) \quad \text{as in Section 3.2.1.}
$$

The dependence of $Z_n(x)$ on $\bar{t}(x)$ is the same regardless of how $\bar{t}(x)$ is computed. Therefore, from the proof of lemma 3.3.2,

$$
\frac{dZ_n(x)}{dt(x)} > 0 .
$$

We conclude that

**Theorem 3.4.1**: In any topological space, and under any general demand distribution, the minisum location coincides with the n-SSFL median.

3.5 Discussion

We have formulated the n-server-single-facility loss system in which we want to locate a single facility on a network in order to minimize system average response time. Servers are indistinguishable as far as their service time distributions are concerned. Server preference structure (or preference probability vector) of calls does not complicate the evaluation of state probabilities. This allows us to use the Erland Loss Formula. We do not have to evaluate state probability to its finest grain. Only the knowledge of the saturation probability is required. Otherwise, we have to resort to numerical methods to compute each state.
probability as discussed in Wolff and Wrighton [80].

We are able to prove the Hakimi nodal result for the n-SSFL median without imposing any restriction on the value of $Q$, the cost of rejection when all servers are busy (except for non-negativity). This is very peculiar since the cost for a lost customer can take on the value zero. This can be seen intuitively as a balance between $t(x)$ and $P_s(x)$. When we place the facility at a very "bad" location, the expected service time increases due to the increase of response time $\bar{t}(x)$. This will increase the saturation probability $P_s(x)$. Even if $Q$ is zero, in which case there is no cost for a lost customer, the increase in $P_s(x)$ (decrease in $1-P_s(x)$) is countered by the deterioration of the response time $\bar{t}(x)$. The final mathematical analysis shows that the best location is still at a place where $\bar{t}(x)$ is minimized.

One can understand this peculiarity (about the cost of rejection $Q$) mentioned above better if one computes the cost to the system per unit time, in the one-server loss system. Consider a very long period of time $T$. The amount of time the server is busy is $P_1 T$, where $P_1 =$ probability that the server is busy. Since the average length of a busy period equals the expected service time $\bar{s}$, there are $P_1 T/\bar{s}$ busy periods in the time span $T$. The cost to the system in one busy period is: (i) $\bar{t}$, the travel time to a service demand that triggers the server to become busy; and (ii) $\lambda \bar{s}Q$, where there are $\lambda \bar{s}$ lost customers, each at a cost $Q$. When the server is idle, it costs the system nothing. Therefore in the time $T$, the cost to the system is (number of busy periods in $T$) x (cost to the system per busy period) = $P_1 T/\bar{s}(\bar{t} + \lambda \bar{s}Q)$, and the cost per unit time is $P_1(\bar{t}/\bar{s} + \lambda Q)$. Since $\bar{s} = \bar{w} + \beta \bar{t}$, $\bar{t}/\bar{s}$ is essentially a constant. Thus, minimizing cost
per unit time is equivalent to minimizing $P_1$, the probability that the server is busy. $P_1$ is minimized if the service time $\bar{s}$ (and therefore $\bar{t}$) is minimized. The above argument is due to Professor Stephen C. Graves of M.I.T.

The availability of preference probability vectors at each node for free servers allows modeling feasibility such as bi-lingual personnel or servers familiar with local neighborhoods. However, the presence of this preference structure does not enter our analysis. The inclusion of such structure is to tie in with previous work, where one has a server preference list at each demand point (e.g., see Larson [44]).

When proving the $n$-server nodal result, we evaluate the sign of $dP_s/dt$. This is seen intuitively (plausible) to be non-negative. The reasoning is as follows: $P_s$ is the probability that all $n$ servers are busy and $\bar{t}(x)$ is the average travel time from $x$. The expected service time is $\bar{s}(x) = \bar{w} + \beta \bar{t}(x)$. $\bar{w}$ is independent of server location. Increasing $\bar{t}(x)$ increases the expected service time for each call. This, in turn, should produce more work for the servers and thus increase the chance that all servers are busy.

We have considered several extensions to this loss model. The first obvious extension is that servers are allowed to reside at different locations. This modification gives rise to different service time distribution for different servers. Hence, the Erlang loss formula does not apply. We also have considered different types of calls to be handled differently by different servers and thus change the non-travel component of total service time to be server specific. This, of course, results in the same difficulty. No existing results on queueing theory can handle such problems.
The nodal result of Berman and Larson [2] assumes the stationarity of steady state probabilities for local change of facility locations. This simplification leads to the nodal locations of servers allowing each to take on different nodes. In our model, the steady state probabilities depend on server location (continuously) through the expected service time. Allowing different locations will result in difficulties discussed earlier.
4.1 Introduction

Ever since Hakimi's work ([26], [27]), there has been considerable interest in the problem of optimally locating one or more facilities on a network. Consider an undirected network $G(N,L)$ with node set $N(\mid N \mid = n)$ and link set $L$, having a fraction $h_j$ of total service demands originating at node $j \in N$. (No demand originates on the links.) If $d(x, j)$ is the distance between the facility at $x \in G$ and node $j \in N$, then the expected travel time associated with a random service demand is

$$
\bar{t}(x) = \sum_{j=1}^{n} h_j d(x,j)/v
$$

where $v$ is the constant travel speed on the network.

Hakimi's one median problem is to locate a facility on the network such that $\bar{t}(x)$ is minimized. Hakimi shows that an optimal location exists in the node set $N$, thus reducing a continuous search to a finite one.

The median problem incorporates only one of the two types of probabilistic behaviors often seen in practice: it does include the probabilistic spatial nature of service demands, using $h_j$ as the probability that a random service demand originates at node $j$; it does not include the probabilistic temporal nature of service demands, which in certain operating systems can result in service demands either being rejected (as studied in Chapter 3) or placed in a queue due to the unavailability of servers. The probability of system saturation (all servers unavailable) is often quite significant; if the server is busy servicing demands 50% of the time, and
if service demands arrive in time in a Poisson manner, then 50% of the arriving service demands find the server busy and are either rejected or placed in queue. With the queueing option, the mean in-queue waiting time can be much larger than the mean travel time, the lone consideration in the median problem. Thus one is motivated to formulate and analyze location problems in which temporal as well as spatial uncertainties are incorporated.

In this chapter, we consider the location on a tree network of a single facility that houses a mobile server. Service demands occur at nodes in a Poisson manner. In response to each demand, the server (if available) travels to the demand to provide on-scene and perhaps off-scene service. If the server is unavailable at the time of a service demand, the demand is entered into a queue that is depleted in a first-in-first-out manner. From a queueing point of view, this is an M/G/1 system (Poisson input, general independent service time with a single server) operating in steady state with infinite queue capacity. The objective is to locate a facility to minimize the sum of mean queueing delay and mean travel time.

Berman, Larson and Chiu [3] examine this problem on a general network and develop a finite step algorithm to obtain the optimal location.

We specialize our analysis on a tree network and introduce an efficient algorithm to "trim" the tree in search of the optimal location. In Section 4.3, we perform parametric analyses on the total Poisson demand rate \( \lambda \) and trace the trajectory of the optimal location when traffic intensity is varied. An effort is made to characterize the optimal value of the objective function as a function of \( \lambda \). We only obtain partial results concerning this value function. To conclude this chapter, we present an
example to show that multiple local optima do exist on a link in a general network. In Chapter 5, we will illustrate the trim algorithm and the parametric analysis in a numerical example with a discrete nodal, as well as continuous link demands.

4.2 Stochastic Queue Median on a Tree Network

4.2.1 Problem Formulation:

Let $G(N,L)$ be an undirected network with node set $N$ ($|N| = n$) and link set $L$. Service demands occur exclusively at the nodes, with each node $j$ generating an independent Poisson stream of rate $\lambda h_j (\sum h_j = 1)$. The travel distance from point $x$ on $G$ to node $i$ in $N$ is $d(x,i)$. The travel distance on link $(i,j)$ is $d_{ij}$. In all cases, travel time is equal to travel distance divided by travel speed $v$. The time required to travel a fraction $\theta$ of link $(i,j)$ is assumed to be $\theta d_{ij}/v$. All travel distances are taken as the shortest distance between two points on the network.

For a facility located at $x$ on $G$, the total service time associated with a service demand is:

$$ \tilde{s}(x) = \tilde{w}_j + \beta/v \ d(x,j) \quad \text{with probability } h_j $$

where $\tilde{w}_j$ is the non-travel component of service time, a random variable. $\beta/v \ d(x,j)$ accounts for round-trip travel time between facility at $x$ and incident at node $j$. Figure 4.2.1 shows the temporal sequence associated with a service demand from node $j$.

We identify the sum of the on-scene and off-scene service time (see Figure 4.2.1) with $\tilde{w}_j$, the non-travel related component of total service time. We let $\bar{w} = \sum_{j \in N} h_j \bar{w}_j$, where $(-)$ indicates expectation of a random variable and $\bar{w}^2 = \sum_{j \in N} h_j \bar{w}_j^2$, $s^2(x) = \text{second moment of service time.}$
service demand occurs from node $j$

arrival of server at node $j$

return to facility

d(x,j)/v

travel time to the scene

on-scene service time

(\$-1)d(x,j)/v

travel time back to facility

off-scene service time

Time

Total service time $S(x)$ with probability $h_j$

Figure 4.2.1 Temporal Sequence of a Service Demand
We assume that $w_j$, $w_j^2$, $w$, and $w^2$ are finite.

Given the facility location at $x$, the expected response time $\overline{TR}(x)$ associated with a random service demand is the sum of the mean in-queue delay $\overline{Q}(x)$ and the expected travel time $\overline{t}(x)$. Since the stochastic system is a single server queue having Poisson input and general independent service time (i.e., an $M/G/1$ system), it is well known that (for example, see Kleinrock [41]):

$$\overline{Q}(x) = \begin{cases} \frac{\lambda_s^2(x)}{2(1-\lambda_s(x))} & \text{for } \lambda_s(x) < 1 \\ +\infty & \text{for } \lambda_s(x) \geq 1 \end{cases}$$

Hence, for $\lambda_s(x) < 1$

$$\overline{TR}(x) = \overline{Q}(x) + \overline{t}(x)$$

where $\overline{t}(x) = \frac{1}{v} \sum_{j \in N} h_{d(x,j)}$, $\overline{s}(x) = \overline{w} + \beta/\nu \overline{t}(x)$. The objective is to find $y \in G(N,L)$ such that

$$\overline{TR}(y) \leq \overline{TR}(x) \quad \text{for all } x \text{ on } G(N,L)$$

Definition: $y$ will be called a Stochastic Queue Median (SQM).

Before we search for this SQM, we will develop an algebraic expression for $\overline{TR}(x)$. With the definition of node partitioning $A(a,b,x)$ and $B(a,b;x)$ (as defined in Chapter 3), when we are at a point $x$ on link $(a,b)$ (i.e., at a point on $(a,b)$ with distance $x$ from node $a$), we can write $\overline{t}(x)$, $\overline{s}(x)$ and $\overline{s^2}(x)$ as follows:
\[ \bar{t}(x) = \frac{1}{v} \left( c_1 x + c_2 \right) \] (4.2.1)

\[ \bar{s}(x) = \bar{w} + \beta \bar{t}(x) = \bar{w} + \frac{\beta}{v} \left( c_1 x + c_2 \right) \] (4.2.2)

\[ \bar{s}^2(x) = \frac{\beta^2}{v^2} x^2 + \left[ \frac{2\beta^2 c_4}{v^2} + \frac{2\beta c_6}{v} \right] x + \left[ \frac{\beta^2}{v^2} + \frac{2\beta}{v} c_7 + \bar{w}^2 \right] \] (4.2.3)

where

\[ c_1 = \sum_{j \in A} h_j - \sum_{j \in B} h_j \]
\[ c_2 = \sum_{j \in A} h_j d(a, j) + \sum_{j \in B} h_j (\ell + d(b, j)) \]
\[ c_3 = \sum_{j \in N} h_j = 1 \]
\[ c_4 = \sum_{j \in A} h_j d(a, j) + \sum_{j \in B} h_j (\ell + d(b, j)) \]
\[ c_5 = \sum_{j \in A} h_j [d(a, j)]^2 + \sum_{j \in B} h_j [\ell + d(b, j)]^2 \]
\[ c_6 = \sum_{j \in A} \bar{w}_j - \sum_{j \in B} h_j \bar{w}_j \]
\[ c_7 = \sum_{j \in A} h_j \bar{w}_j d(a, j) + \sum_{j \in B} h_j \bar{w}_j [d(b, j) + \ell] \]

Note that the generic expression for \( \bar{s}^2(x) \) is:

\[ \bar{s}^2(x) = \sum_{j \in N} h_j \left[ \bar{w}_j + \frac{\beta}{v} d(x, j) \right]^2 \]
In a general network, the sets A and B change as one moves from node a to node b (i.e., as x increases). Specifically, A will shrink and B will grow correspondingly. We will recall two definitions given in Chapter 3.

**Definition:** A breakpoint is a point on a link at which the set A (B) changes.

**Definition:** A primary region is an interval on any link for which the sets A and B remain unchanged.

We note that the sets A and B remain unchanged on a link in a tree network. Thus the entire link is a primary region in a tree network. We will first study the behaviors of $\overline{t}(x)$, $\overline{s^2}(x)$ and $\overline{TR}(x)$ in a primary region. In Section 4.2.3, we will examine the properties of $\overline{t}(x)$, $\overline{s^2}(x)$ and $\overline{TR}(x)$ on a path of a tree network. We present an efficient search procedure by trimming away portions of a tree in determining the location of the stochastic queue median in Section 4.2.4.

**4.2.2 Behavior of $\overline{t}(x)$, $\overline{s^2}(x)$ and $\overline{TR}(x)$ in a Primary Region**

All the results in this section are valid for a general network. Before we begin our analysis, we want to inform the readers that the algebra involved is intrinsically complicated. We have tried our best to reduce all expressions to their simplest possible forms. Instead of explicitly expressing the dependence of the equations on x (location of the facility on G), we use $\overline{t}(x)$, $\overline{s(x)}$, $\overline{s^2}(x)$ and their derivatives in all the equations. The dependence on x becomes implicit through $\overline{t}(x)$, $\overline{s(x)}$, $\overline{s^2}(x)$, etc., and this reduces the algebra from a possible two-page equation to a two-line one. The beauty of most of the analyses lies in the fact that we fix the
optimal location at x on G, and ask the question: "What values of \( \lambda \) will make x optimal?". Therefore, in the analysis that follows (throughout the thesis), the reader can consider \( s^2, s, t, s^2', s' \), etc., as numbers rather than functions of x. The only relevant information are the signs of these functions, which will be apparent as we move along. The first two results concern the behavior of \( t \) and \( s^2 \).

Lemma 4.2.1: \( t(x) \) is linear in a primary region.

Proof: Since \( t(x) = \frac{1}{v} (c_1x + c_2) \).

Lemma 4.2.2: \( s^2(x) \) is convex in a primary region.

Proof: \[
\frac{d^2(s^2(x))}{dx^2} = 2 \beta^2 / v^2 > 0.
\]

Before we analyze the behavior of \( TR(x) \) in a primary region, we will state an important inequality which will aid future analysis, looking first at the generic expression of \( s^2(x) \), for x on \((a,b)\).

\[
s^2(x) = \sum_{j \in A} h_j [w_j + \beta/v(x + d(a,j))]^2 + \sum_{j \in B} [w_j + \beta/v(d(b,j) + \lambda - x)]^2
\]

Note that for node partitions \( A(a,b; x) \) and \( B(a,b; x) \), \( s^2(x) \) is defined through the parameters c's, and they are defined and valid over only one primary region. However, we observe that \( s^2(x) \) is non-negative for all values of \( x \in R \). Thus, we can extend the domain of definition of \( s^2(x) \) and note that this quadratic expression of equation 4.2.3 is non-negative for all values of \( x \). Therefore the value of \( s^2(0) \) and \( s^2'(0) \) (denoted by \( s_0^2 \) and \( s_0^2' \)) are well defined even if the primary region over which the sets A and B are partitioned does not contain the point \( x = 0 \). We can
write $s^2(x)$ as:

$$s^2(x) = \beta^2/v^2 x^2 + s_0^2 + s_0^2 > 0$$

for all $x \in \mathbb{R}$

and thus

$$\left(\frac{s_0^2}{s_0^2}\right) - 4 \beta^2/v^2 s_0^2 < 0$$

Algebraic manipulation also reveals that:

$$\left(\frac{s^2(x)}{s_0^2}\right)^2 - 4 \beta^2/v^2 s^2(x) = \left(\frac{s_0^2}{s_0^2}\right)^2 - 4 \beta^2/v^2 s_0^2 < 0$$

which is independent of $x$.

With these observations, we are ready to prove a convexity result of $\overline{TR}(x)$.

**Lemma 4.2.3**: $\overline{TR}(x)$ is convex in a primary region over which $\overline{TR}(x)$ is finite (i.e., $1 - \lambda \overline{s}(x) > 0$).

**Proof**: We evaluate the second derivative of $\overline{TR}(x)$ directly.

$$\frac{d^2 TR(x)}{dx^2} = \frac{\lambda \beta}{v} \left[ 1 - \lambda s \right]^{-3} \left[ \left( \frac{\beta}{v} s - c_1 s s^2 + \beta \overline{s} c_1 \overline{s} \right) \lambda^2 + \left( c_1 s^2 - 2\beta \overline{s} \right) \beta + \beta \overline{s} \right]$$

where we have suppressed the argument $x$.

We are only interested in the case where $(1 - \lambda \overline{s}(x)) > 0$. Therefore, $\frac{d^2 TR(x)}{dx^2}$ takes on the sign of the numerator, which is a quadratic equation in $\lambda; \text{ we will call it } N(\lambda)$. We note that $N(0) = \beta/v > 0$. Let $N(\lambda) = \gamma_1 \lambda^2 + \gamma_2 \lambda + \gamma_3$ and compute
\[
\gamma_2^2 - 4\gamma_1\gamma_3 = c_1^2 [(s_0^2) - 4s_0^2/v^2 s_0^2]
= c_1^2 [(s_0^2) - 4s_0^2/v^2 s_0^2] < 0
\]

by expression (4.2.4). We conclude that \(N(\lambda)\) has no real roots and thus \(N(0) > 0\) implies \(N(\lambda) > 0\) for all values of \(\lambda\). Therefore, \(\frac{d^2TR(x)}{dx^2} > 0\), and convexity of \(\overline{TR}(x)\) follows.

4.2.3 Properties of \(\overline{t}(x), \overline{s}(x), \overline{TR}(x)\) over Paths of a Tree Network

Since the entire link in a tree network is a primary region, we know that \(\overline{t}(x)\) (linear), \(\overline{s}(x)\) and \(\overline{TR}(x)\) are all convex over the entire link in a tree network. We would like this convexity behavior to carry over when we move across nodes from one link to another. This turns out to be true. Before we present our result, we will examine the change of the node partitions \(A\) and \(B\) as one moves across a node.

Suppose we are moving along a path from node \(i\) to node \(j\) to node \(k\) (or \(i - j - k\)) in a tree. From now on, we will not mention the context of tree network.

![Diagram of tree network with nodes i, j, and k, and partitions A', B', A'', B'', C', and C'']

We will denote the respective node partitions \(A\), \(B\) and constants \(C\) as \(A'\), \(B'\) and \(C'\); for link \((i,j)\), \(A''\), \(B''\) and \(C''\) for link \((j,k)\).

**Definition:** \(\Delta(j; i,k) = \) set of nodes (including \(j\)) connected to node \(j\) after the removal of links \((i,j)\) and \((j,k)\). When no ambiguity arises, we will simply use \(\Delta\).
It is easily verified that

\[ A'' = A' \cup \Delta \]
\[ B' = B'' \cup \Delta \]
\[ A', \Delta, B'' \text{ are mutual disjoint.} \]

In a tree network, where the sets A and B remain unchanged on a link, derivatives of \( TR(x) \) (when finite), \( s^2(x) \), \( \bar{s}(x) \), and \( \bar{t}(x) \) exist except at the nodes. (Note that the value of these functions is continuous, when finite, at the nodes.) To facilitate discussion, we will introduce the notion of in and out derivatives of a function at a node along a path. When we are moving along a path \( i-j-k \), the in (or out) derivative of a function at node \( j \) is evaluated with parameters (the sets A, B and constants \( C_s \)) relevant to the link \((i,j) \) (or \((j,k)\)). We will now investigate the properties of \( s^2(x) \), \( \bar{s}(x) \), \( \bar{t}(x) \) and \( TR(x) \) along a path in a tree.

**Lemma 4.2.4:** The function \( s^2(x) \) is convex on any path in a tree network.

**Proof:** Since \( s^2(x) \) is convex on a link, it suffices to show that along any path \( i-j-k \)

\[ s^2_j \text{ in } \leq s^2_j \text{ out} \]
Let the length of \((i,j)\) be \(d_{ij}\).

\[
\bar{s}^2(j)_{\text{in}} = 2\beta^2/v^2 x + 2\beta^2/v^2 c_4 + 2\beta/v c_6 \mid x=d_{ij}
\]

\[
= 2\beta^2/v^2 d_{ij} + 2\beta^2/v^2 c_4 + 2\beta/v c_6
\]

and

\[
\bar{s}^2(j)_{\text{out}} = 2\beta^2/v^2 x + 2\beta^2/v^2 c_4 + 2\beta/v c_6 \mid x=0
\]

\[
= 2\beta^2/v^2 c_4 + 2\beta/v c_6
\]

We can show that

\[
c_6'' = \sum_{\lambda \in A''} h_{\lambda} \overline{w}_{\lambda} - \sum_{\lambda \in B''} h_{\lambda} \overline{w}_{\lambda}
\]

\[
= \sum_{\lambda \in A'} h_{\lambda} \overline{w}_{\lambda} + \sum_{\lambda \in \Delta} h_{\lambda} \overline{w}_{\lambda} - \left[ \sum_{\lambda \in B'} h_{\lambda} \overline{w}_{\lambda} - \sum_{\lambda \in \Delta} h_{\lambda} \overline{w}_{\lambda} \right]
\]

\[
= \sum_{\lambda \in A'} h_{\lambda} \overline{w}_{\lambda} - \sum_{\lambda \in B'} h_{\lambda} \overline{w}_{\lambda} + 2 \sum_{\lambda \in \Delta} h_{\lambda} \overline{w}_{\lambda}
\]

\[
= c_6' + 2 \sum_{\lambda \in \Delta} h_{\lambda} \overline{w}_{\lambda}
\]

A similar argument and the facts that

(i) \(d(i,\lambda) - d_{ij} = d(j,\lambda)\) for \(\lambda \in \Delta\) and

(ii) \(d(j,\lambda) = d(k,\lambda) + d_{jk}\) for \(\lambda \in B''\)

give

\[
c_4'' = c_4' + d_{ij} + 2 \sum_{\lambda \in \Delta} h_{\lambda} d(j,\lambda)
\]
With \(c'_4, c'_6\) expressed in terms of \(c'_4\) and \(c'_6\), we can relate \(s'^2(j)_{\text{out}}\) to \(s'^2(j)_{\text{in}}\):

\[
\frac{s'^2(j)_{\text{out}}}{s'^2(j)_{\text{in}}} = 1 + \frac{4\beta^2/v^2 \sum_{\lambda \in \Delta} h_{\lambda}^d(j, \lambda) + 4\beta/v \sum_{\lambda \in \Delta} h_{\lambda}^w_{\lambda}}{c_1/v^2 T_{\lambda}^h_{\lambda}(\lambda, j, k)}
\]

Therefore \(s^2(x)\) is convex along any path in a tree network.

**Lemma 4.2.5:** \(\bar{t}(x)\) is convex along any path in a tree network.

**Proof:** Since \(t(x)\) is linear on a link, it suffices to show that along path \(i-j-k\)

\[
\bar{t}(j)_{\text{in}} \leq \bar{t}(j)_{\text{out}}
\]

\[
\bar{t}(j)_{\text{in}} = c'_1/v \quad \bar{t}(j)_{\text{out}} = c''_1/v
\]

but

\[
c''_1 = c'_1 + 2 \sum_{j \in \Delta} h_j \geq c'_1
\]

\[
\therefore \bar{t}(j)_{\text{in}} \leq \bar{t}(j)_{\text{out}}
\]

**Lemma 4.2.6:** \(\bar{s}(x)\) is convex along any path in a tree network.

**Proof:** Since \(\bar{s}(x) = \bar{w} + \beta \bar{t}(x)\), and \(\beta > 1\), the result follows.

**Lemma 4.2.7:** \(\bar{TR}(x)\) is convex along any path in a tree network when it is finite.
Proof: Again convexity of $\overline{\text{TR}}(x)$ over a link permits us to prove this lemma by showing that along $i-j-k$ at node $j$:

$$\overline{\text{TR}}(j)' \geq \frac{\lambda}{2} \left[ 1 - \lambda s(j) \right]^{-2} \left[ s(j)' \right]_{\text{out}}^2 \left( 1 - \lambda s(j) \right) + \lambda s(t(j))' \left[ s(j)' \right]_{\text{out}}^2 + t(j)'$$

since $1 - \lambda s(j) > 0$

$$s(j), \ s(j)'' \quad \text{are continuous at node } j$$

$$s(j)'_{\text{out}} \geq s(j)'_{\text{in}}$$

and

$$t(j)'_{\text{out}} \geq t(j)'_{\text{in}}$$

$$\therefore \overline{\text{TR}}(j)'_{\text{out}} \geq \overline{\text{TR}}(j)'_{\text{in}}$$

For a convex function along any path of a tree network, we can prove the following lemma.

**Lemma 4.2.8:** Suppose $F(x)$ is convex along any path of a tree network and $F(i) < F(j)$ where $(i,j)$ is a link; then for all $x$ on $(j,k), k \neq i, F(j) < F(x)$.

**Proof:** Convexity of $F$ along path $i-j-k$ (in particular) and the fact that $F(i) < F(j)$ implies that

$$F(j)'_{\text{out}} > 0 \quad \text{along any path } (j,k), k \neq i.$$
Again using convexity of $F$ along a link, we conclude that $F(x) > F(j)$ for all $x \in (j,k)$, $k \neq i$.

Since $\overline{TR}(x)$, $\overline{s^2}(x)$, $\overline{t}(x)$ and $\overline{s}(x)$ are convex along any path in a tree, lemma 4.2.8 applies to all of them. We will use this property to eliminate portions of a tree in search of the Stochastic Queue median in the next section.

4.2.4 An Efficient Algorithm to Locate the Stochastic Queue Median of a Tree

Since $\overline{TR}(x)$ is convex along any path of the tree, we can start from any end node (a node with degree one, i.e., number of links incident on it is one) and follow a link with decreasing derivatives at each node encountered in the path. However, we can take advantage of the structure of $\overline{TR}(x)$ and do significantly better. Recall that

$$\overline{TR}(x) = \frac{\lambda \overline{s^2}(x)}{2(1-\lambda \overline{s}(x))} + \overline{t}(x) \quad \text{when } 1-\lambda \overline{s}(x) > 0.$$ 

where

$$\overline{s}(x) = \overline{w} + \beta \overline{t}(x)$$

If we start from any end, $1 - \lambda \overline{s}(x)$ could be negative. In that case, convexity of $\overline{TR}(x)$ does not hold and we are left with no direction. If we start at a point $x$ where $1 - \lambda \overline{s}(x)$ is positive and examine a direction of movement; suppose we find that both $\overline{t}(x)$ and $\overline{s^2}(x)$ are increasing in that direction, we can abandon our search along that direction. This is because $\overline{t}(x)$ and $\overline{s^2}(x)$ are convex -- if they start increasing they will keep on increasing. This also causes $1 - \lambda \overline{s}(x)$ to decrease and approach
the value zero. The overall effect is the increase of $\overline{TR}(x)$ as long as $1 - \lambda \overline{s}(x)$ is positive. If $1 - \lambda \overline{s}(x)$ ever becomes zero, the convex nature of $\overline{s}(x)$ tells us to stop searching.

The above observation prompts us to start our search at a point where $\overline{TR}(x)$ is finite, i.e., $1 - \lambda \overline{s}(x) > 0$. The most obvious starting point is the Hakimi median where $\overline{t}(x)$ (and thus $\overline{s}(x)$) is at its minimum. Any movement away from the Hakimi median will result in the continued increasing of $\overline{t}(x)$ as guaranteed by the convexity of $\overline{t}(x)$. Thus, one needs only to check the value of $\overline{s}^2(x)$ (or its derivative) for direction of movement. This line of reasoning leads us to develop an efficient algorithm to locate the Stochastic Queue Median (SQM).

Before starting the algorithm, we will introduce some standard network notations and definitions. Again, we are dealing exclusively with undirected and connected tree networks:

Definition: A tree is rooted at node $i$ if we define the depth of a node $j$, $d_i(j)$, as the number of links between nodes $i$ and $j$, and we call node $i$ the root of the tree.

For two nodes $p$, $q$ in a tree rooted at node $i$, we have the following definition:

Definition: $p$ is the predecessor of $q$ if $d_i(p) < d_i(q)$ (or the immediate predecessor if $d_i(p) = d_i(q) - 1$) and $d(p,q) = d(i,q) - d(i,p)$; where $d(m,n)$ is the shortest distance between nodes $m$ and $n$ on the network, $q$ is the successor (or immediate successor) of $p$.

We now study the behavior of $\overline{t}(x)$ on a tree network rooted at the Hakimi Median (HM).
Lemma 4.2.9: Suppose \( i \) is the Hakimi Median on a tree network. Then \( c_1 \) is non-negative on all links \((i,j)\). (Note that when we say link \((i,j)\), a direction is imposed on the movement of \( x \) from \( i \) to \( j \).)

Proof: The HM minimizes \( t(x) = \frac{1}{v}(c_1 x + c_2) \) over the network, implying that \( x = 0 \) minimizes \( c_1 x + c_2 \) along all links \((i,j)\) where \( c_1 \) and \( c_2 \) are parameters on link \((i,j)\); \( c_1 > 0 \) follows.

Lemma 4.2.10: \( c_1 > 0 \) on \((p,q)\) where \( q \) is an immediate successor of \( p \) in a tree rooted at the Hakimi median.

Proof: The parameter \( c_1 > 0 \) on \((i,j)\) where \( i \) is the Hakimi median, together with convexity of \( \bar{t}(x) \) along any path of a tree network, guarantees the positivity of \( c_1 \) on \((p,q)\), since \( c_1 \) is seen to be the derivative of \( \bar{t}(x) \) along \((p,q)\).

From the above analyses, we know that when one moves away from the Hakimi median through nodes of increasing depth, the function \( \bar{t}(x) \) (and thus \( \bar{s}(x) \)) increases. Therefore, we need only check on the behavior of \( \bar{s}^2(x) \) as a direction of search down the rooted tree. (Moving down a rooted tree means moving along a path of increasing depth.) There are many variations of this testing mechanism. We can test the value of \( \bar{s}^2 \) directly, or we can check the sign of \( \bar{s}^2 \). In what follows, we will investigate each possibility. First, we define a branch of a rooted tree.

Definition: Branch \( k \) of a rooted tree is the sub-tree consisting of node \( k \), its successors and the links connecting them.
Theorem 4.2.1: Consider a tree rooted at the Hakimi Median, i. Suppose
\( s^2(j) > s^2(i) \), where \( j \) is an immediate successor of \( i \). We can eliminate
branch \( j \) in determining the stochastic queue median (note that we cannot
ignore link \((i,j)\)).

Proof: Because \( TR(x) \) is convex along any path of a tree, it is sufficient
to show that \( TR(j)' \) along \((i,j)\) is positive.

\[
TR(j)'_{in} = \frac{\lambda}{2} [1 - \lambda s(j)]^{-2} [s^2(j)' (1 - \lambda s(j)) + \lambda \beta \frac{c_1}{v} s^2(j)] + \frac{c_1}{v}
\]

There are two cases to consider:

1. \( 1 - \lambda s(j) \leq 0 \), \( TR(j)' \) is infinite, convexity of \( s(j) \)
and \( s(j)'_{in} = \beta/v c_1 \geq 0 \) tells us that \( 1 - \lambda \ s(x) \) will
never become positive again, further down the rooted
tree.

2. \( 1 - \lambda s(j) > 0 \). We know \( c_1 \geq 0 \) down the rooted tree.
All we have to show is \( s^2(j)'_{in} \geq 0 \). This is true
because \( s^2(i) < s^2(j) \) and \( s^2(x) \) is convex along link
\((i,j)\).

Thus, we can eliminate branch \( j \) from consideration.

Corollary: Consider a tree rooted at the Hakimi median \( i \), and consider
a particular sequence of nodes along a path from \( i \): \( i-j-k-q \). Suppose
\( s^2(i) > s^2(j) > s^2(k) \), and \( s^2(k) < s^2(q) \). Then we can eliminate branch
\( q \) from consideration (refer to Figure 4.2.2).

Proof: On link \((k,q)\), \( c_1 > 0 \) (lemma 4.10) implies \( t(q)'_{in} = \frac{1}{v} c_1 > 0 \) on
link \((k,q)\). The exact argument, as in Theorem 4.1, gives the desired
result.
Figure 4.2.2 A Tree rooted at node i
As noted in Theorem 4.2.1 and its corollary, we can eliminate branch k but not the link from node k pointing towards the Hakimi median. The following lemma allows us to do so selectively.

**Lemma 4.2.11:** Eliminate link (j,k), j being the immediate predecessor of k in a Hakimi median rooted tree if $s^2(j)'_{\text{out}} > 0$.

**Proof:** $s^2(x)' > 0$ for all x on (j,k) by convexity of $s^2(x)$; $t(x)' = c_1/v > 0$ by lemma 4.10. These two conditions imply that $\overline{TR}(x)$ is increasing along (j,k).

In addition to the above elimination procedures, we also have to check whether $1 - \lambda \overline{s}(x)$ is positive to ensure the finiteness of $\overline{TR}(x)$. We want to know the structure of the "trimmed" tree after the application of these elimination processes. The discussion that follows will be informal, but mathematically justifiable. The informal discussion will be followed by a theorem describing the efficiency of the trimming procedure.

We will first describe our trimming procedure.

**The SQM Trimming Algorithm (on a tree network):**

1. Root the tree at the Hakimi Median.
2. Evaluate $s^2(j)'_{\text{out}}$ at each node, in order of increasing depth, along link (j,k) where k is the immediate successor of j. Stop if j is an end of the tree.
3. Eliminate link (j,k) and branch k if $s^2(j)'_{\text{out}} > 0$.

There are several observations we wish to make regarding this algorithm. The first one is the choice of test quantity $s^2(j)'_{\text{out}}$ instead of $s^2(j)_{\text{out}}$ at each node. The reason for this is that we can eliminate not only branch k but also link (j,k) if $s^2(j)'_{\text{out}} > 0$ along link (j,k). This eliminates the possibility of links "sticking" out from a node. Also,
the evaluation of $s^2(j)'_{\text{out}}$ is a little easier than $s^2(j)$ since only $c_4$ and $c_6$ need be computed (see Equation 4.2.3 for the expression of $s^2(x)'$ evaluated at $x = 0$ along link $(j,k)$). The second observation concerns streamlining of this trimming procedure. Suppose we are now at node $j$ and $j(k)$, $k = 1, 2, \ldots, \ell$ are the immediate successors of $j$ (see Figure 4.2.3). Due to the convexity of $s^2(x)$ along any path in a tree network, we see that at most one $s^2(j)'_{\text{out}}$ along links $(j,j(k))$, $k = 1, 2, \ldots, \ell$, can be negative.

The next observation deals with justification of this algorithm. This trimming procedure does not exclude any optimal solution to our problem because: (1) at node $j$, we eliminate portions of a tree with larger $s^2$ guaranteed by $s^2(j)'_{\text{out}} \geq 0$ and convexity of $s^2$ down the tree; (2) convexity of $\bar{t}(x)$ along a path and the fact that we eliminate portions of a tree (rooted at the Hakimi median) "deeper" than node $j$ assure us that $\bar{t}(x)$ ($\bar{s}(x)$) will be at least as large as $\bar{t}(j)$ if we go further down the tree. Therefore, $\overline{\text{TR}}(j)$ is no bigger than any $\overline{\text{TR}}(x)$ associated with the eliminated sub-tree. Note that we have not concerned ourselves with the finiteness of $\overline{\text{TR}}(x)$ in our trimming operation. Also the value of $\lambda$ has not entered into our consideration. We could evaluate $\overline{\text{TR}}(j)'_{\text{out}}$ at each node $j$. Convexity of $\overline{\text{TR}}(x)$ will guarantee the same operations and justifications as using $s^2(x)'$. However, we will be computing a more complicated expression (involving $c_1, c_2, c_5, c_7$ and $\lambda$) and we will not be taking advantage of the rooting (at the Hakimi median) operation. The fact that we move away from the Hakimi median allows us not to check the value of $\bar{t}(x)$.

With the above observations and the trimming algorithm, we conclude:
Figure 4.2.3 A Step in the Trimming Algorithm
Theorem 4.2.2 (Efficiency of the Trimming Algorithm): The trimmed tree consists of a single path with nodes of increasing depth leading out of its root, the Hakimi median.

After the trimming operation, we are left with a sub-tree which consists of a single path leading out of the Hakimi median. The following search procedure, applied to the residual tree, using the convex property of $\overline{TR}(x)$, locates the Stochastic Queue Median:

Search Procedure:

1. Evaluate $1 - \lambda \overline{s}(j)$, and $\overline{TR}(j)'$ at node $j$ along this path in order of increasing depth.

2. If $k$ is the first node such that either (i) $1 - \lambda \overline{s}(k) \leq 0$ or (ii) $\overline{TR}(k)'_{\text{out}} > 0$, the SQM is located on link $(\ell,k)$, where $\ell$ is the immediate predecessor of $k$ in this path.

3. If all $1 - \lambda \overline{s}(j) > 0$ and $\overline{TR}(j)'_{\text{out}} < 0$ (or $\overline{TR}(j)'_{\text{in}} < 0$ if $j$ is the last node on the path) on this path, then the last node is the SQM.

Optimality of the above procedure is guaranteed by the convexity of $\overline{TR}(x)$ along any path in a tree. We still need to specify the operation in Step (2), namely, how to locate the SQM on link $(\ell,k)$. There are three possibilities.

1. $\overline{TR}(k)'_{\text{in}} < 0$ (note $\overline{TR}(k)'_{\text{out}} > 0$) and $1 - \lambda \overline{s}(k) > 0$.

   In this case, node $k$ is the SQM due to the convexity of $\overline{TR}(x)$.

2. $1 - \lambda \overline{s}(k) > 0$ and $\overline{TR}(k)'_{\text{in}} > 0$ along link $(\ell,k)$. When this occurs, we have to find the minimum of $\overline{TR}(x)$ on
on link \((\ell,k)\) by examining the derivative of \(\overline{TR(x)}\) with respect to \(x\):

\[
\frac{d\overline{TR(x)}}{dx} = \frac{1}{2} [1 - \lambda \overline{s}(x)]^{-2} [B_1(\lambda)x^2 + B_2(\lambda)x + B_3(\lambda)]
\]

where

\[
B_1(\lambda) = -\frac{\beta^2}{v^3} c_1 (\beta - 2c_1^2) \lambda^2
\]

\[
B_2(\lambda) = -2\beta/v^2 (\beta - 2c_1^2)(\overline{w} + \beta/v c_2)\lambda^2 + 2\beta/v^2 (\beta - 2c_1^2)\lambda
\]

\[
B_3(\lambda) = \left[\frac{\beta}{v} c_1 (\beta^2/v^2 c_5 + 2\beta/v c_7 + \overline{w^2}) + \frac{2}{v} c_1 (\overline{w} + \beta/v c_2)\lambda^2 - 2\beta/v (\beta/v c_4 + c_6)(\overline{w} + \beta/v c_2)\lambda^2 + [2\beta/v (\beta/v c_4 + c_6) - \frac{4}{v} c_1 (\beta/v c_2 + \overline{w})]\lambda + 2/v c_1
\]

To find the minimum of a convex function, we set \(\frac{d\overline{TR(x)}}{dx} = 0\) and solve for \(x\) (note \(1 - \lambda \overline{s}(x) > 0\) for \(x \in [0,m]\), where \(m\) is the length of link \((\ell,k)\), \(G(x) = B_1(\lambda)x^2 + B_2(\lambda)x + B_3(\lambda) = 0\). We know \(\overline{TR(\ell)}_{\text{out}} < 0\) and \(\overline{TR(k)}_{\text{in}} > 0\). Therefore, we have \(G(0) < 0\) and \(G(m) > 0\). This implies \(G(x) = 0\) has a solution for \(x \in [0,m]\). We have to make some observations before solving this quadratic equation in \(x\):

First, \(c_1 > 0\) on this path;

Second, we will show in the next section that \(\beta - 2c_1^2 > 0\) on this path,

Third, \(B_2(\lambda) = 2\beta/v^2 (\beta - 2c_1^2)\lambda (1 - \lambda \overline{s}(\ell)) > 0\),

Fourth, \(B_3(\lambda)\) takes on the sign of \(\frac{d\overline{TR(x)}}{dx} \big|_{x=0}\) which is negative.

We will show that \(x\) takes on the \((+)\) root in this quadratic solution \(G(x) = 0\), that is,
We will consider both situations in which \( c_1 \) can take on positive or negative values, since in a general network, \( c_1 \) can be negative. In the case that \( c_1 = 0 \),

\[
x = -\frac{B_3(\lambda)}{B_2(\lambda)}.
\]

(i) \( c_1 > 0 \), which is our current situation on a tree network: \( B_1(\lambda) = -\beta^2/\nu^3 \ c_1(\beta - 2c_1^2)\lambda^2 < 0 \) implies \( G(x) \) is concave. \( G(0) < 0, \ G(m) > 0 \) allows us to sketch \( G(x) \) in Figure 4.4(a). Note that \( B_2(\lambda) > 0, \ B_3(\lambda) < 0, \) and

\[
x_1 = \frac{-B_2(\lambda) + \sqrt{B_2(\lambda)^2 - 4B_1(\lambda)B_3(\lambda)}}{2B_1(\lambda)} < \frac{-B_2(\lambda) - \sqrt{B_2(\lambda)^2 - 4B_1(\lambda)B_3(\lambda)}}{2B_1(\lambda)} = x_2
\]

Therefore, referring to Figure 4.2.4(a), we want to take the (+) root, that is, \( x_1 \).

(ii) When \( c_1 < 0 \), which is possible in a general network, \( B_1(\lambda) > 0 \) implies \( G(x) \) is convex, and we again have \( B_2(\lambda) = \frac{dG(x)}{dx} \bigg|_{x=0} > 0, \ B_3(0) = G(0) < 0, \ G(m) > 0. \) From the sketch of \( G(x) \) in Figure 4.2.4(b), we note

\[
x_2 = \frac{-B_2(\lambda) - \sqrt{B_2(\lambda)^2 - 4B_1(\lambda)B_3(\lambda)}}{2B_1(\lambda)} < 0
\]

Therefore, we want \( x_1 = \frac{-B_2(\lambda) + \sqrt{B_2(\lambda)^2 - 4B_1(\lambda)B_3(\lambda)}}{2B_1(\lambda)} \), the only feasible choice.
Figure 4.2.4 Sketch of $G(x)$
(3) The last situation is $1 - \lambda \overline{s}(k) < 0$. We know $1 - \lambda \overline{s}(\ell) > 0$ since node $k$ is the first node such that $1 - \lambda \overline{s}(k) < 0$. This implies $\overline{s}(k) > \overline{s}(\ell)$ or $c_1 > 0$ (since $\overline{s}(k) = \overline{s}(\ell) + c_1 x \big|_{x=m}$). Also we know that there exists an $x \in (0, m)$ such that $1 - \lambda \overline{s}(x) = 0$. We will denote this $x$ by $x_s$.

Simple substitution of $x_s = \frac{[1 - \lambda (\overline{w} + \beta/v c_2)]/\lambda c_1}{\lambda c_1}$ reveals that $G(x_s) > 0$, $G(0) < 0$, $G(x_s) > 0$ and Roll's Theorem implies that $G(x)$ passes through the $x$ axis somewhere between 0 and $x_s$. The same observation as in case (2) (Figure 4.2.4(a)) tells us that we should take the $(+)$ root in the quadratic solution $G(x) = 0$.

The above trimming and search procedures are very efficient since we trim away a large portion of the tree. To locate the Hakimi median in a tree requires (worst case) $O(n)$ operations (where $n$ is the number of nodes in a tree). In the Trim Algorithm, we have to evaluate $\overline{s}(j)'_{out}$ at each depth of the rooted tree. However, we do not have to evaluate this for all the links of the tree due to the fact that at most, one $\overline{s}(j)'_{out}$ can be negative. This again requires $O(n)$ operations. The search procedure is applied to the residual tree, which again requires no more than $O(n)$ operations. Therefore, the worst case complexity is order $n$. However, we expect the average performance to be much better. We will quantify the above discussion next.

We can make some very rough estimate of the average behavior of the Trim and Search Algorithms. Suppose we are given a tree with $n$ nodes and the average degree of each node (except its ends) is $d$. After we root the tree at the Hakimi median, we expect there are $\ell$ levels in the rooted
tree (we count the Hakimi median as being at level one, and its immediate successors at level two, etc.). n, d, and \( \ell \) are related roughly as

\[
\sum_{j=0}^{\ell-1} (d-1)^j = \frac{(d-1)^{\ell} - 1}{d-1} = \frac{(d-1)^{\ell} - 1}{d-\ell} = n
\]

or

\[
\ell = \ell n_{d-1}[n(d-2)+1]
\]

At each level, we only have to consider one node, and at most one link leading out of it. We also check the sign of \( s^2(i)_{\text{out}} \) at most \( (d-1) \) times.

We conclude that we will perform on the average (and at most)

\[
(\ell-1)(d-1) = (d-1)[\ell n_{d-1}[n(d-2)+1] - 1] \text{ operations.}
\]

As for the search algorithm, the number of operations is no more than the number of links in the residual path, which in our average analysis is about \( \ell - 1 \), or \( \ell n_{d-1}[n(d-2)+1] - 1 \) times. To illustrate our analysis, we consider a tree with an average degree of three (except its ends). This tree will be like a binary tree, as illustrated in Figure 4.2.5. For concreteness, suppose this tree has 15 nodes.

\[
d = 3, \quad n = 15 \quad \text{imply} \quad \ell = \ell n_2[15(3-2)+1] = 4
\]

\[
(\ell-1)(d-1) = 3 \times 2 = 6 \quad \text{operations}
\]

An exhaustive search will require searching over all 14 links. The search algorithm requires about \( \ell - 1 = 3 \) computations.
Figure 4.2.5  A Binary Tree with Four Levels
As another example, consider $d = 4$, $n = 121$. In this case, the average level is $\lambda = 4$ and the number of operations is $(4-1)(4-1) = 9$ for the trim operations. For the search algorithm, it is $4 - 1 = 3$ operations. We would like to point out that each operation here involves the evaluations of the coefficients $c's$ in the expressions for $s^2(x)$ and $TR(x)$. To locate the Hakimi median, it requires $n$ operations, but each operation involves only the update of nodal weights $h_j$. The saving is in the reduction of operations involving computation of $TR(x)$ and $s^2(x)$, even though it takes $n$(simple) operations in locating the Hakimi median.

One final remark concerning nodal restriction is in order before we turn to the topic of parametric analysis. Due to the convex nature of $TR(x)$, if we impose the additional restriction that the facility must be located on a node, we need only check on node $\lambda$ and node $k$ (refer to Step (2) of the search procedure).

It turns out that the trim and search procedures described here are also applicable to the stochastic queue one median problem defined on a tree network with continuous link demands (see Chapter 5). We will illustrate the procedures with a numerical example at the end of Chapter 5.

4.3 Parametric Analysis (on $\lambda$) of the SQM Location

4.3.1 Preliminary

In locating the SQM in a tree, we only have to follow a path of decreasing $s^2(x)$ starting from the Hakimi median (HM). This is due to the positivity of $c_1$ when one moves away from the HM. Once $s^2(x)' becomes positive, all points down the path from the HM will never be candidates for the SQM regardless of the value of $\lambda$. This is a very crude test and uses an elementary observation of the functional form of $TR(x) = \lambda s^2(x)/$
2(1 - \lambda \overline{s}(x)) + \overline{t}(x). As a matter of fact, one can eliminate branches of the tree before \overline{s^2}(x)' becomes positive. We intend to develop a stronger and \lambda-independent test quantity here (as is the test quantity \overline{s^2}(x)' , contrasted to the evaluation of \overline{TR}(x)' which depends on the value of \lambda). The development and analysis of such a test quantity not only sheds light on proof techniques that follow, but also is desirable when we try to find the locations of all the SQM's in a tree (i.e., the SQM for all values of \lambda). In other words, the development in this section will set the stage for parametric analysis of \lambda.

Before we start to find such a test quantity, we will introduce some notations and make several observations. We are operating under two general constraints: (i) \overline{s^2}(x)' < 0 and (ii) 1 - \lambda \overline{s}(x) > 0; this limits the range of \lambda for fixed value of x. \overline{s^2}(x)' < 0 means that we are interested only in points on the median seeking path (MSP) -- a path starting from the HM in the direction of decreasing \overline{s^2}(x). We note that the MSP is independent of \lambda and is unique. As before, x on (i,j) represents a point on link (i,j) at a distance x from node i. A point y down the path from x means that x is in the unique path between the HM and the point y. For convenience, we re-label the nodes along the MSP as (i_1, i_2, ...), where i_1 = HM. The letters i, j, k will denote nodes of the network. We will write \overline{TR}(x) as \overline{TR}(x,\lambda) to highlight the dependence of \overline{TR} on \lambda as well as on x. Our intention is to test where a point x on G will ever be a stochastic queue median, for any value of \lambda.

Definition: \text{DTX}(x,\lambda) = \frac{\partial \overline{TR}(x,\lambda)}{\partial x} = \text{derivative of } \overline{TR} \text{ with respect to } x.

When x is at node i, we use the notations \text{DTX}(i,\lambda)_\text{in} and \text{DTX}(i,\lambda)_\text{out} to denote the in and out derivatives of \overline{TR}(x,\lambda) (with respect to x) at node i.
along the implied path, MSP. From the expression for \( TR(x,\lambda) \), we obtain:

\[
DTX(x,\lambda) = \frac{\lambda s^2(x)'(1-\lambda s(x)) + \lambda^2 \beta/v c_1 s^2(x)}{2(1-\lambda s(x))} + \frac{1}{v} c_1 = \frac{A(x,\lambda)}{2(1-\lambda s(x))^2}
\]

where

\[
A(x,\lambda) = [\beta/v c_1 s^2(x) + \frac{2}{v} c_1 (s(x))^2 - s(x)s^2(x)']\lambda^2 + [s^2(x) - \frac{4c_1}{v} s(x)]\lambda + \frac{2}{v} c_1 = A_1(x)\lambda^2 + A_2(x)\lambda + A_3.
\]

Since we are interested only in those \((\lambda,x)\) pairs such that \(1-\lambda s(x) > 0\), we know \(DTX(x,\lambda)\) and \(A(x,\lambda)\) have the same signs. If at the point \(x\), \(DTX(x,\lambda)\) is positive for all values of \(\lambda\) we can conclude that \(x\) will never be a candidate for the SQM regardless of the value of \(\lambda\). This is due to the convexity of \(TR(x,\lambda)\) as a function of \(x\). Our aim now is to examine the sign of \(A(x,\lambda)\) for fixed \(x\) and varying \(\lambda\). We realize that \(A(x,\lambda)\) is quadratic in \(\lambda\) (when \(x\) is fixed), and there is a well established rule to test the sign of a quadratic function. First we will examine the coefficients of \(\lambda^2\), and \(\lambda\) in \(A(x,\lambda)\).

We are now on a path of decreasing \(s^2\) and increasing \(t\); therefore, \(s^2' < 0\) and \(t' = c_1 > 0\), and

\[
A_1(x) > 0
\]
\[
A_2(x) < 0
\]
\[
A_3 > 0
\]
\[
A(x,0) = A_3 > 0
\]
\[
\left. \frac{A(x,\lambda)}{\partial \lambda} \right|_{\lambda=0} = A_2(x) < 0
\]
and \( \frac{\partial^2 A(x,\lambda)}{\partial \lambda^2} = 2A_1(x) > 0 \) for all values of \( \lambda \).

**Lemma 4.3.1:** \( A(x,\lambda) \) is convex in \( \lambda \) for fixed values of \( x \) along the MSP and has a global minimum at \( \lambda = \frac{A_2(x)}{2A_1(x)} \). Figure 4.3.1 shows a sketch of \( A(x,\lambda) \) for fixed \( x \).

**Lemma 4.3.2:** \( A(x,\lambda) \), for \( x \) on MSP, has no zero in \( \lambda \) iff \( A_2(x)^2 - 4A_1(x)A_3 < 0 \).

**Lemma 4.3.2** reduces to the following inequality:

\[
8\beta/v^2 \, c_1 \frac{\partial^2 s(x)}{\partial x^2} - (s^2(x)')^2 > 0
\]

This condition is the same as:

\[
\min_\lambda A(x,\lambda) > 0
\]

and

\[
\min_\lambda A(x,\lambda) = 8\beta/v^2 \, c_1 \frac{\partial^2 s(x)}{\partial x^2} - (s^2(x)')^2 > 0.
\]

Again, \( s^2(x)' \) is negative, since we are on the MSP. Note that \( s^2(x)' < 0 \) does not guarantee \( 8\beta/v^2 \, c_1 \frac{\partial^2 s(x)}{\partial x^2} - s^2(x)' > 0 \). We can view this as a second level test above \( s^2(x)' < 0 \), and it is \( \lambda \)-independent. If this test quantity is positive, we know that at \( x,\overline{TR}(x)' \) is always positive and thus will never be a candidate for the SQM.

**Theorem 4.3.1:** Along the MSP, we can eliminate all points \( y \) down the path of \( x \) as candidates for the SQM if

\[
8\beta/v^2 \, c_1 \frac{\partial^2 s(x)}{\partial x^2} - (s^2(x)')^2 > 0.
\]
Figure 4.3.1 Sketch of $A(x, \lambda)$ in $\lambda$
This result is useful in obtaining all the SQM in a tree when all possible values of \( \lambda \) are considered.

The following remarks are in order:

- Strictly speaking, we have to consider \( \text{SIGN} (\min_{\lambda} DTX(x, \lambda)) \) for \( x \) on the MSP. However, assuming \( 1 - \lambda \bar{s}(x) > 0 \), i.e., restricting the range of \( \lambda \) to \( \lambda < 1/s(x) \), implies

\[
\text{SIGN} (\min_{\lambda} DTX(x, \lambda)) = \text{SIGN} (\min_{\lambda} A(x, \lambda)).
\]

- We have to be sure that, at each point \( x \), the restriction on the range of \( \lambda \) (i.e., \( 1 - \lambda \bar{s}(x) > 0 \)) presents no problems in our analysis. This will become apparent in the next section.

- \( DTX(x, \lambda) \) and \( A(x, \lambda) \) have the same zeros in \( \lambda \) in the range of interest (\( \lambda < 1/s(x) \)).

We will now turn to the parametric analysis of \( \lambda \).

4.3.2 Parametric Analysis of \( \lambda \) on the Location of the SQM

We wish to locate all the SQM's of a tree, in the sense that we want to find the location of the SQM as a function of \( \lambda \). We will introduce the notion of the \( DTX(x, \lambda) \) profile in \( \lambda \)-space and present a graphic way to trace the locus of the SQM as \( \lambda \) varies. A by-product of this analysis is to find a range of \( \lambda \) such that the HM coincides with the SQM.

Berman, Larson and Chiu [3] show that as \( \lambda \) approaches zero or \( \lambda_{\text{max}} \), where \( \lambda_{\text{max}} \) is the value of \( \lambda \) such that \( 1 - \lambda \bar{s}(i) = 0 \), \( i \) being the HM, the SQM coincides with the HM. They further observe (computationally) that the SQM moves away from the HM as \( \lambda \) increases, and then retreats.
towards the HM upon further increase of $\lambda$. Their computational experience shows that the locus of the SQM (as a function of $\lambda$), in a general undirected network, may not even be continuous. We will quantify such observations in the case of a tree network. We again remind the reader that we are on the MSP, i.e., on point $x$, such that $s_2(x)' < 0$ and $c_1 > 0$.

Lemma 4.3.3: For fixed values of $x$, $DTX(x, \lambda)$ has a unique minimum in $(0, 1/s(x))$.

Proof: Recalling $DTX(x, \lambda) = A(x, \lambda)/2[1 - \lambda s(x)]^2$, we have

1. $DTX(x,0) = A_3/2 > 0$,
2. $A(x, 1/s(x)) = A_1(x)[1/s(x)] + A_2(x)[1/s(x)] + A_3 = (\beta/v)c_1(s^2(x))/[s(x)]^2 > 0$,
3. $DTX(x,0) < DTX(x, \lambda)$, as $\lambda + 1/s(x)$ from below,
4. $\frac{\partial DTX(x, \lambda)}{\partial \lambda}\big|_{\lambda=0} = 1/2 s^2(x)' < 0$,
5. $\frac{\partial DTX(x, \lambda)}{\partial \lambda} = 0$ solving for $\lambda$ gives $\lambda_c = -s^2(x)' / [2 \beta/v c_1 s^2(x) - s(x) s^2(x)'] > 0$,
6. $\lambda_c < \frac{1}{s(x)} : \frac{-s^2(x)'}{2\beta/v c_1 s^2(x) - s(x) s^2(x)'} < \frac{1}{s(x)}$

Points (i) to (v) above show that $DTX(x, \lambda)$, as a function of $\lambda$ at fixed $x$, has a unique minimum at $\lambda_c \in (0, 1/s(x))$. ■

Figure 4.3.2 shows two different profiles of $DTX(x, \lambda)$ at $x_1$ and $x_2$ in $\lambda$-space. Note that due to the convexity of $TR(x, \lambda)$ in $x$, the profile at a point $y$ down the path of $x$ will be enveloped below by the profile of $x$. In Figure 4.3.2, we know that $x_2$ is down the path of $x_1$. As will be seen later, $x_1$ and $x_2$ belong to two different links of the MSP. $\lambda_c$ is the value of $\lambda$ at which $DTX(x, \lambda)$ achieves its minimum. We now state a
Figure 4.3.2  DTX(x,\lambda) Profiles at x_1 and x_2
Lemma 4.3.4: \( \min_{\lambda} \text{DTX}(x,\lambda) > 0 \) iff \( \frac{8\beta/v^2 c_1^2 s^2(x) - (s^2)'^2}{2(1-\lambda_c s(x))^2 (2\beta/v c_1 s^2(x) - s^2(x) s(x))} > 0 \).

Proof: \( \min_{\lambda} \text{DTX}(x,\lambda) = \text{DTX}(x,\lambda_c) = \frac{8\beta/v^2 c_1^2 s^2(x) - (s^2)'^2}{2(1-\lambda_c s(x))^2 (2\beta/v c_1 s^2(x) - s^2(x) s(x))} \).

Since \( 1 - \lambda_c s(x) > 0 \) as seen in Lemma 4.3.3 and \( s^2(x)' < 0, c_1 > 0 \), we have \( \text{SIGN} \left[ \min_{\lambda} \text{DTX}(x,\lambda) \right] = \text{SIGN} \left[ \frac{8\beta/v^2 c_1^2 s^2(x) - (s^2)'^2}{2(1-\lambda_c s(x))^2 (2\beta/v c_1 s^2(x) - s^2(x) s(x))} \right] \).

We will make some remarks before stating the main result of this section. For fixed values of \( x \):

- \( \text{DTX}(x,0) = \text{DTX}(y,0) = 1/v c_1 \) if \( x \) and \( y \) are on the same link,

- \( \text{DTX}(x,\lambda) \) is supported below by \( \text{DTX}(y,\lambda) \) for all values of \( \lambda \) of interest if \( x \) is down the path of \( y \). By "\( \lambda \) of interest", we mean the range of \( \lambda \) such that \( 1 - \lambda_c s(x) > 0 \) in the associated profile \( \text{DTX}(x,\lambda) \).

- \( \text{DTX}(x,\lambda) \) and \( A(x,\lambda) \) have the same zeros in \( \lambda \) for fixed \( x \).

- Along the MSP, we know that \( \frac{\overline{(s^2)'^2} - 8\beta/v^2 c_1^2 s^2}{2} > 0 \) by our \( \lambda \)-independent test, and \( \frac{\overline{(s^2)'^2} - 4\beta^2/v^2 s^2}{2} < 0 \) by the proof of lemma 4.2.3. The above inequalities imply that \( \beta - 2c_1^2 > 0 \).

Theorem 4.3.2: the SQM of a tree network moves (as a function of \( \lambda \)) down the MSP from the HM continuously and then retreats continuously along the MSP back to the HM upon further increase of \( \lambda \).
Proof: First we will examine the movement of the SQM when it is on the interior of a link. We know that for each point x on the MSP, there are associated values of λ for which x is the SQM. Specifically, the (x, λ) pair satisfies \( A(x, \lambda) = 0 \). We will investigate the movement of x by examining the derivative of the SQM location with respect to λ.

\[
\frac{dx}{d\lambda} = x' = \frac{-\partial A / \partial \lambda}{\partial A / \partial x}
\]

\[
\partial A / \partial x = 2\beta \nu^2 \lambda (\beta - 2c_1^2)(1 - \lambda s) > 0
\]

because \( \beta - 2c_1^2 > 0 \) from Section 4.3.2 and \( 1 - \lambda s > 0 \) since x is optimal. Therefore \( x' \) takes on the sign of \(-\partial A / \partial \lambda\) at the point where \( A(x, \lambda) = 0 \).

We know that \( A(x, \lambda) \) is convex in \( \lambda \) (quadratic) for fixed values of x. Also, \( \partial A / \partial x > 0 \) implies that when \( \lambda \) is fixed, \( A(x_1, \lambda) < A(x_2, \lambda) \) if \( x_1 < x_2 \) (\( x_1 \) and \( x_2 \) on the same link).

We now refer our discussion to Figure 4.3.3, which shows the graphs of \( A(x_1, \lambda) \) and \( A(x_2, \lambda) \), where \( x_1 \) and \( x_2 \) are fixed and on the same link and \( x_1 < x_2 \). At \( \lambda = 0 \), \( A(x_1, 0) = A(x_2, 0) = 2/\nu c_1 \). These two curves (as a function of \( \lambda \)) will not intersect because \( \partial A / \partial x > 0 \) (except at \( \lambda = 0 \)).

Both of them will cross the \( \lambda \)-axis because \( x_1 \) and \( x_2 \) are on the MSP (lemma 4.3.2). \( x' = dx/d\lambda \) -- the derivative of the SQM as a function of \( \lambda \) -- is positive at \( \lambda^-(x_1) \), \( \lambda^-(x_2) \) and negative at \( \lambda^+(x_1) \) and \( \lambda^+(x_2) \). In other words, the SQM is advancing down the MSP at \( \lambda^-(\cdot) \) and retreating back to the HM at \( \lambda^+(\cdot) \).

At a node \( j \), \( A(j, \lambda) \) depends on whether one is going into node \( j \) or moving away from node \( j \). Exactly the same argument we use to prove convexity of \( TR(x) \) across a node applies here: \( A(j, \lambda)_{\text{out}} \) is enveloped below
Figure 4.3.3 $A(x, \lambda)$ at two points, $x_1$ and $x_2$, on the same link.

Figure 4.3.4 $A(x, \lambda)$ at node $j$. 
below by $A(j,\lambda)_{in}$ as shown in Figure 4.3.4. We note that $A(j,0)_{in} < A(j,0)_{out}$ because the parameter $c_1$ increases across a node. For $\lambda \in A(j^-)$ or $A(j^+)$, node $j$ remains the SQM. This is because, for $\lambda \in A(j^-)$ or $A(j^+)$, $A(x,\lambda) > 0$ for $x$ down the path of node $j$ and $A(y,\lambda) < 0$ for $y$ up the path of node $j$; and since $DTX(x,\lambda)$ takes on the same sign as $A(x,\lambda)$, no interior (to a link) point $x$ will satisfy $A(x,\lambda) = 0$, and therefore, node $j$ is the SQM.

As one moves down the MSP, either of the following two situations will occur: (i) there is a point $x_m$ ($x_m$ could be a node) such that $\left(s'_{in}\right)^2 - 88/v^2 c_1 \frac{2}{s_{in}^2} = 0$; or (ii) there is a node $j$, such that $\left(s'_{in}\right)^2 - 88/v^2 c_1 \frac{2}{s_{in}^2} > 0$ and $\left(s'_{out}\right)^2 - 88/v^2 c_1 \frac{2}{s_{out}^2} < 0$. Both situations are illustrated in Figures 4.3.5 (case ii) and Figures 4.3.6 (case i), where we have sketched the $DTX(x,\lambda)$ profiles instead of $A(x,\lambda)$ to highlight the critical value at saturation (when $1 - \lambda s = 0$).

For each location $x$ along the MSP, there exist $\lambda^-(x)$ and $\lambda^+(x)$, as in Figure 4.3.3, where $x'$ at $\lambda^-(x)$ is positive and negative at $\lambda^+(x)$.

Same observations can be made when $x$ is a node. In that case, $x'$ is positive at $\lambda^-(j)_{in}$ and $\lambda^-(j)_{out}$, and negative at $\lambda^+(j)_{in}$ and $\lambda^+(j)_{out}$.

With the above argument, we conclude that:

(i) **Case 1:** When there is a point $x_m$ on the MSP such that $\left(s'_{in}\right)^2 - 88/v^2 c_1 \frac{2}{s_{in}^2} = 0$, $A(x_m,\lambda)$ or $DTX(x_m,\lambda)$ touches the $\lambda$-axis at $\lambda_m$ (see Figure 4.3.6):

$$x' > 0, \text{ for } \lambda \leq \lambda_m; x' < 0, \text{ for } \lambda \geq \lambda_m.$$

(ii) **Case 2:** When there is a node $j$ on the MSP such that $\left(s'_{in}\right)^2 - 88/v^2 c_1 \frac{2}{s_{in}^2} > 0$ and $\left(s'_{out}\right)^2 - 88/v^2 c_1 \frac{2}{s_{out}^2} < 0$, $A(j,\lambda)_{in}$ or $DTX(j,\lambda)_{in}$ intersects the $\lambda$-axis at
\( \lambda^{-}(j)_{in} \) and \( \lambda^{+}(j)_{in} \):

\[ x' > 0, \text{ for } \lambda \leq \lambda^{-}(j)_{in}; \quad x' < 0, \text{ for } \lambda \geq \lambda^{+}(j)_{in}. \]

We know that the MSP is independent of \( \lambda \) (as long as (a) \( 1 - \lambda \bar{s} > 0 \),

(b) \( s'^{2} < 0 \), (c) \( (s'^{2})^{2} - 8\beta/\nu^{2} c_{1}^{2} \bar{s}^{2} > 0 \)). The above argument shows

that the SQM moves along the MSP away from the HM as \( \lambda \) increases and

retreats back to the HM along the MSP as \( \lambda \) increases further. We illustrate the movement of the SQM in Figures 4.3.5 and 4.3.6.

In Figure 4.3.5, we have highlighted five \( \text{DTX}(x, \lambda) \) profiles: (1) \( \text{DTX}(i_{1}, \lambda)_{out} \); (2) \( \text{DTX}(i_{2}, \lambda)_{in} \); (3) \( \text{DTX}(i_{2}, \lambda)_{out} \); (4) \( \text{DTX}(i_{3}, \lambda)_{in} \); (5) \( \text{DTX}(i_{3}, \lambda)_{out} \). The shaded region corresponds to points on the same link (in this case, link 1 is bracketed by profiles (1) and (2), link 2 is bracketed by profiles (3) and (4)). The correspondence between the value of \( \lambda \) and the location of the SQM is as follows:

<table>
<thead>
<tr>
<th>( \lambda )</th>
<th>Location of SQM</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 ( \rightarrow \lambda_{1} )</td>
<td>( i_{1} ) (HM)</td>
</tr>
<tr>
<td>( \lambda_{1} \rightarrow \lambda_{2} )</td>
<td>points on link 1 (( i_{1} \rightarrow i_{2} ))</td>
</tr>
<tr>
<td>( \lambda_{2} \rightarrow \lambda_{3} )</td>
<td>( i_{2} )</td>
</tr>
<tr>
<td>( \lambda_{3} \rightarrow \lambda_{4} )</td>
<td>points on link 2 (( i_{2} \rightarrow i_{3} ))</td>
</tr>
<tr>
<td>( \lambda_{4} \rightarrow \lambda_{5} )</td>
<td>( i_{3} )</td>
</tr>
<tr>
<td>( \lambda_{5} \rightarrow \lambda_{6} )</td>
<td>points on link 2 (( i_{3} \rightarrow i_{2} ))</td>
</tr>
<tr>
<td>( \lambda_{6} \rightarrow \lambda_{7} )</td>
<td>( i_{2} )</td>
</tr>
<tr>
<td>( \lambda_{7} \rightarrow \lambda_{8} )</td>
<td>points on link 1 (( i_{2} \rightarrow i_{1} ))</td>
</tr>
<tr>
<td>( \lambda_{8} \rightarrow 1/s(i_{1}) )</td>
<td>( i_{1} )</td>
</tr>
<tr>
<td>( \lambda \rightarrow 1/s(i_{1}) )</td>
<td>no finite value of ( \overline{TR}(x) ) exists</td>
</tr>
</tbody>
</table>
Each shaded area corresponds to DTX profiles of points on the same link.

Figure 4.3.5 DTX Profiles and Trajectory of SQM.
A different scenario is that the SQM moves back toward the HM in the interior of a link. This happens if there exists a point $x_m$ on the MSP such that $A_2(x_m)^2 - 4A_1(x_m)A_3 = 0$. This situation is shown graphically in Figure 4.3.4, which is analogous to Figure 4.3.5. Note that at $x_m$, $A_2(x_m)^2 - 4A_1(x_m)A_3 = 0$.

The four DTX profiles are: (1) DTX$(i_1,\lambda)_\text{out}$; (2) DTX$(i_2,\lambda)_\text{in}$; (3) DTX$(i_2,\lambda)_\text{out}$; (4) DTX$(x_m,\lambda)$. The trajectory of the SQM as a function of $\lambda$ is:

<table>
<thead>
<tr>
<th>$\lambda$</th>
<th>Location of SQM</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0 + \lambda_1$</td>
<td>$i_1 = \text{HM}$</td>
</tr>
<tr>
<td>$\lambda_1 + \lambda_2$</td>
<td>link 1 $(i_1 \to i_2)$</td>
</tr>
<tr>
<td>$\lambda_2 + \lambda_3$</td>
<td>$i_2$</td>
</tr>
<tr>
<td>$\lambda_3 + \lambda_4$</td>
<td>point of link 2 $(i_2 \to x_m)$</td>
</tr>
<tr>
<td>$\lambda_4$</td>
<td>$x_m$</td>
</tr>
<tr>
<td>$\lambda_4 + \lambda_5$</td>
<td>point of link 2 $(x_m \to i_2)$</td>
</tr>
<tr>
<td>$\lambda_5 + \lambda_6$</td>
<td>$i_2$</td>
</tr>
<tr>
<td>$\lambda_6 + \lambda_7$</td>
<td>link 1 $(i_2 \to i_1)$</td>
</tr>
<tr>
<td>$\lambda_7 + 1/s(i_1)$</td>
<td>$i_1$</td>
</tr>
<tr>
<td>$\lambda &gt; 1/s(i_1)$</td>
<td>no finite value of $\text{TR}(x)$ exists on the network</td>
</tr>
</tbody>
</table>

Intuitively, we are just picking the zeros of DTX$(x,\lambda)$ in $\lambda$ as $x$ changes. Convexity of $\text{TR}(x,\lambda)$ in $x$ gives layers of DTX profiles with no intersection, except possibly at $\lambda = 0$. This property disallows zig-zagging behavior of the SQM as $\lambda$ varies.

There is a special case in which $c_1 = 0$ along one link of the MSP. Figure 4.3.7 shows the MSP from node 1 to node 2, and so on. Node 1 is
Each shaded area corresponds to DTX profiles of points on the same link.

Figure 4.3.6 DTX Profiles and Trajectory of SQM
Figure 4.3.7 The MSP Rooted At Node 1

Figure 4.3.8 DTX(x,λ) Profiles With Multiple Minisum Locations.
the Hakimi median; $c_1$ along link (1,2) is zero. The implication is that we have degeneracy associated with the HM. Node 1, along with node 2 and all points on link (1,2), are all minisum locations. The $DT(x,\lambda)$ profiles are shown in Figure 4.3.8.

Profile (a) represents $DT(x,\lambda)_{\text{out}}$, where $x$ is node 1.
Profile (b) represents $DT(x,\lambda)_{\text{in}}$, where $x$ is node 2.
Profile (c) represents $DT(x,\lambda)_{\text{out}}$, where $x$ is node 2.
Profile (d) represents $DT(x,\lambda)_{\text{in}}$, where $x$ is node 3.

In this situation, the MSP begins with node 2 instead of node 1; all analysis remains identical from node 2 on.

A by-product of these observations is that we can find the range of $\lambda$ such that the HM coincides with the SQM.

**Lemma 4.3.5**: Let $i$ be the HM. Then $i_0 = \text{SQM}$ for $0 \leq \lambda \leq \lambda^-$ and $\lambda^+ \leq \lambda \leq 1/s(i_1)$ where $\lambda^- < \lambda^+$, and they are the roots of $A(i_1,\lambda) = 0$.

4.3.3 A Numerical Example

To give the reader a feel for $\lambda^-$ and $\lambda^+$, we have constructed a numerical example as shown in Figure 4.3.9. The weights, $h_j$'s, are shown beside the nodes; the link lengths are shown next to the arcs. The Hakimi median is found to be at node 2. We have computed the following relevant parameters for $A(x,\lambda)$ for $x$ at node 2.

$$\overline{s^2} = 343.88$$
$$s_1^2 = -43.096$$
$$\overline{s}_{\text{out}} = 13.432$$
$$c_1 = 0.066$$

for $\beta = 2, \overline{w} = 1, v = 1, \overline{w^2} = 1.$
Figure 4.3.9  A Numerical Example To Find $\bar{\lambda}$ and $\lambda^*$
A(x, λ) is: 648.07288λ^2 - 46.642λ + 0.132 = 0

or \( \lambda^- = 0.002951 \), \( \lambda^+ = 0.0690192 \)

also \( \frac{1}{\delta} = 0.074449 \)

Therefore, for \( \lambda \in [0, 0.002951] \cup [0.0690192, 0.74449) \), the Hakimi median, i.e., node 2, is the SQM.

4.4 The Optimal Value Function in \( \lambda \)

4.4.1 Preliminary

We define the optimal value function \( v(\lambda) \) as follows:

\[
v(\lambda) = \min_{x \in G} \overline{TR}(x, \lambda)
\]

That is, the optimal average response time of the Stochastic Queue Median problem as a function of the arrival rate \( \lambda \). There have been few results concerning the behavior of the expected waiting time (as a function of the number of servers, for example) in the queueing literature. Besides general theoretical interest, we can use \( v(\lambda) \) to study the uniform pruning problem: Upon arrival of a call for service, Bernoulli trial is performed to decide whether this call is to be serviced by a secondary unit (at a cost \( Q \) with probability \( p \)). One can balance the cost \( Q \) against the marginal change of \( \overline{TR}(x, \lambda) \) to obtain an optimal Bernoulli probability. Due to the Poisson nature of the arrival process, the resulting M/G/1 system is modified only in its arrival rate, from \( \lambda \) to \( (1-p)\lambda \). This uniform pruning problem can be stated as follows:

\[
\min_{p \in [0,1]} (1-p)v((1-p)\lambda) + pQ
\]
If $v(\lambda)$ is convex, the uniform pruning problem is a convex programming problem. One needs only to find a $p$ satisfying the first order optimality condition. We are able to prove the convexity of $v(\lambda)$ for a certain range of $\lambda$. However, we conjecture the convexity of $v(\lambda)$ for all $\lambda$ of interest; i.e., for $0 \leq \lambda < 1/s(i)$, where $i$ is the Hakimi median. We will first introduce some notations and definitions.

**Definition:** We define the Median Seeking Path (MSP) as a path leading out of the Hakimi median along the direction of $x$ such that: (i) $s''(x) < 0$; and (ii) $(s''(x))' \cdot 88/v^2 c_1 s''(x) > 0$. Note that along this directed path, $c_1 \geq 0$.

For convenience, we will re-label the node along the MSP in increasing order, from the Hakimi median. Refer all definitions to Figures 4.4.1 (a) and (b).

**Definition:** $\Lambda(i)$ is the set of values of $\lambda$ such that node $i$ is the SQM.

Referring to Figures 4.4.1(a) and (b), $\Lambda(i)$ consists of either (i) two non-overlapping intervals as in (a), or (ii) sometimes one continuous interval as in (b). In the case of (a), we can write $\Lambda(i)$ as

$$\Lambda(i) = \Lambda(i^-) \cup \Lambda(i^+)$$

where

$$\lambda^- \leq \lambda^+ \text{ for all } \lambda^- \in \Lambda(i^-) \text{ and } \lambda^+ \in \Lambda(i^+)$$

**Definition:** $\Lambda(i, i+1) = \text{the value of } \lambda \text{ such that points on link } (i, i+1) \text{ become the SQM continuously from node } i \text{ to node } i+1, \text{ as } \lambda \text{ increases}.$

$\Lambda(i+1, i) = \text{the value of } \lambda \text{ such that points on link } (i+1, i) \text{ become the SQM continuously from node } i+1 \text{ to node } i \text{ as } \lambda \text{ increases}.$
Figure 4.4.1 (a) Range of $\lambda$ and DTX Profiles.
Figure 4.4.1 (b) Range of $\lambda$ and DTX Profiles.
We want to point out that
\[ \lambda_1 \leq \lambda_2 \quad \text{for all } \lambda_1 \in \Lambda(i, i+1) \]
and \[ \lambda_2 \in \Lambda(i+1, i) \]

Assume node 1 is the HM, and the nodes are labelled as (1, 2, 3, ..., R) along the MSP. In Figure 4.4.1(a), we define \( x_m = R \), the interior point on a link, such that the SQM moves back towards the HM when \( \lambda \) increases. In this case, \( \Lambda(R) = \Lambda(x_m) = \{\lambda_m\} \).

**Definition:** \( \Lambda \equiv \) the set of values of \( \lambda \) such that the SQM problem is feasible, i.e., \( \Lambda = [0, 1/s(1)] \)

\[
\Lambda = [\bigcup_i \Lambda(i)] \cup [\bigcup_i \Lambda(i, i+1)] \cup [\bigcup_i \Lambda(i, i-1)]
\]

where each of its components are non-overlapping intervals, and

\[ \Lambda(i) = \Lambda(i^-) \cup \Lambda(i^+) \quad \text{for } i \neq R \]

\( \Lambda(R) \) consists either of one point or one single interval.

**4.4.2 Properties of \( v(\lambda) \)**

With all the notations and definitions introduced, we can state some properties of \( \lambda(v) \).

**Lemma 4.4.1:** \( v(\lambda) \) is continuous in \( \lambda \) for all \( \lambda \in \Lambda \).

**Proof:** for \( \lambda \in \Lambda(i) = \Lambda(i^-) \cup \Lambda(i^+) \), \( v(\lambda) = \frac{\lambda s^2(i)}{2(1-\lambda s(i))} + \frac{1}{v} \tilde{t}(i) \), which is continuous in \( \lambda \). For \( \lambda \in \Lambda(i, i+1) \) or \( \lambda \in \Lambda(i+1, i) \), \( v(\lambda) = \frac{\lambda s^2(x)}{2(1-s(x))} + \frac{1}{v} \tilde{t}(x) \), where \( (x, \lambda) \) satisfy \( \Lambda(x, \lambda) = 0 \). As seen in Section 4.2.4 --
Search Algorithm -- \( A(x,\lambda) \) is quadratic in \( x \), \( A(x,\lambda) = B_1(\lambda)x^2 + B_2(\lambda)x + B_3(\lambda) \) and
\[
x^* = \frac{-B_2(\lambda) + \sqrt{B_2(\lambda)^2 - 4B_1(\lambda)B_3(\lambda)}}{2B_1(\lambda)}
\]

Since \( \lambda < 1/s(x) \) as discussed in Section 4.3, \( v(\lambda) \) is continuous in \( \lambda \).
(Note that \( x^* \) is continuous in \( \lambda \).) At the boundary (i.e., where the coefficients defining \( s^2(x) \), \( t(x) \), and \( s(x) \) change -- all the c's) at which \( \lambda = \Lambda(i) \cap \Lambda(i, i+1) \) and \( \lambda = \Lambda(i^+) \cap \Lambda(i, i-1) \), we have no problems because \( s^2(x) \), \( t(x) \), and \( s(x) \) are continuous at the nodes (going in and coming out).

Lemma 4.4.2: \( v(\lambda) \) is continuously differentiable for all \( \lambda \in \Lambda \).

Proof: We will present two proofs, first algebraically, and then, utilizing the fact that \( A(x,\lambda) = 0 \), functionally.

We first define \( v'(\lambda) = \frac{dv}{d\lambda} \), \( x' = \frac{dx}{d\lambda} \) through \( A(x,\lambda) = 0 \).

Proof One: For \( \lambda \in \Lambda(i^+) \) or \( \Lambda(i^-) \); node \( i \) is the SQM,
\[
. \quad \frac{dv}{d\lambda} = \frac{s^2(i)}{2(1-\lambda s(i))^2} \quad \text{continuous in } \lambda.
\]

For \( \lambda \in \Lambda(i, i+1) \), suppressing argument \( x \):
\[
\frac{dv}{d\lambda} = \frac{(1-\lambda s) \left[ 1-s^2 x' + s^2 \right] + \lambda s^2 \left[ \frac{\beta}{\nu c_1 x' + s} \right] + \frac{1}{\nu c_1} x'}{2(1-\lambda s)^2} =
\]
\[
= \frac{s^2}{2(1-\lambda s)^2} \left[ \frac{\lambda^2 (\beta/\nu c_1 s^2 - s^2 s'^2) + s^2 \lambda + c_1/\nu 2(1-\lambda s)^2}{2(1-\lambda s)^2} \right] =
\]
\[ \frac{\overline{s^2}}{2(1-\lambda \overline{s})^2} + x' A(x, \lambda) = \frac{\overline{s^2}}{2(1-\lambda \overline{s})^2} \]

Since \( \overline{s^2} \), \( \overline{s} \) are continuous at the nodes, \( v'(\lambda) \) is continuous also at the boundary (node) \( \Rightarrow v(\lambda) \) is continuously differentiable for \( \lambda \in \Lambda \).

**Proof Two:** For \( \lambda \in \Lambda(i, i+1) \),
\[
\frac{dv}{d\lambda} = \frac{\partial v}{\partial \lambda} + \frac{\partial v}{\partial x} \frac{dx}{d\lambda} = \frac{\overline{s^2}}{2(1-\lambda \overline{s})^2} + \frac{\partial TR(x, \lambda)}{\partial x} x'
\]
but \( \frac{\partial TR(x, \lambda)}{\partial x} = DX(x, \lambda) = 0 \) (since \( DX \) and \( A \) have the same zeros).

\[
\therefore v'(\lambda) = \frac{\overline{s^2}}{2(1-\lambda \overline{s})^2},\ \text{differentiability of } v(\lambda)
\]
follows. \( \blacksquare \)

**Lemma 4.4.3:** \( v(\lambda) \) is increasing for \( \lambda \in \Lambda \).

**Proof:** \( \therefore v'(\lambda) = \frac{\overline{s^2}}{2(1-\lambda \overline{s})^2} > 0. \) \( \blacksquare \)

Without evaluating the derivatives of \( v(\lambda) \), we can prove lemma 4.4.3 directly using an entirely different argument.

**Alternate Proof of Lemma 4.4.3:** We want to show that \( \lambda_2 > \lambda_1 \Rightarrow v(\lambda_2) > v(\lambda_1) \). Let \( x_1 \) and \( x_2 \) minimize \( TR(x, \lambda_1) \) and \( TR(x, \lambda_2) \) respectively over a tree network. This means:

\[
1 - \lambda_1 \overline{s(x_1)} > 0 \quad 1 - \lambda_2 \overline{s(x_2)} > 0 \\
\lambda_2 > \lambda_1 \quad \Rightarrow \quad 1 - \lambda_1 \overline{s(x_2)} > 0
\]

or \( x_2 \) is feasible in minimizing \( TR(x, \lambda_1) \). Optimality of \( x_1 \) in minimizing \( TR(x, \lambda_1) \) implies
All we need to show now is

\[ \nu(\lambda_2) = T(x_2, \lambda_2) > T(x_2, \lambda_1) \]

or

\[ \frac{\lambda_2 s^2(x_2)}{2(1-\lambda_2 s(x_2))} > \frac{\lambda_1 s^2(x_2)}{2(1-\lambda_1 s(x_2))} \]

which is evident because \( \lambda_1 < \lambda_2 \).

\[ \therefore \quad \nu(\lambda_2) > \nu(\lambda_1). \]

**Lemma 4.4.4:** \( \nu(\lambda) \) is convex for \( \lambda \in \Lambda(i) \cup \Lambda(i^+) \).

**Proof:** For \( \lambda \in \Lambda(i) \cup \Lambda(i^+) \), node \( i \) remains optimal.

\[ \therefore \quad \nu''(\lambda) = \frac{d^2 \nu(\lambda)}{d\lambda^2} = \frac{s(i) s^2(i)}{(1-\lambda) s(i)^3} > 0. \]

Convexity follows.

To prove convexity of \( \nu(\lambda) \) on all \( \lambda \in \Lambda \), we still have to show that \( \nu(\lambda) \) is convex over \( \Lambda(i, i+1) \) and \( \Lambda(i+1, i) \), since we have shown convexity of \( \nu(\lambda) \) in \( \Lambda(i) \) and that \( \nu(\lambda) \) is continuous. We have not been successful in this respect. However, we will evaluate the functional form of \( \nu''(\lambda) \) over \( \Lambda(i, i+1) \) and \( \Lambda(i+1, i) \).

We have shown that (suppressing argument \( x \)) \( \nu'(\lambda) = \frac{1}{2} s^2(1-\lambda s)^{-2} \).

Recalling that \( x \) is the interior of a link and \( (x, \lambda) \) satisfy \( \Lambda(x, \lambda) = 0 \), therefore,
\[ x' = \frac{dx}{d\lambda} = \frac{-3A}{3A} \frac{d\lambda}{dx} \]

and

\[ v''(\lambda) = \frac{1}{2} (1 - \lambda s) \left[ 2s - x' \left( (1 - \lambda s)^{2} + 2 \beta/v \c_1 s^{2} \right) \right] \]

and

\[ x' = (-3A/\partial \lambda)(\partial A/\partial x)^{-1} = \frac{-2(\beta/v \c_1 s^{2} - \frac{2}{v} c_1 (s)^{2} - \frac{2}{s} s^2') \lambda + (s^2' - 4c_1/v s)}{2\beta/v^2 \lambda (\beta-2c_1^2)(1-\lambda s)} \]

Substitution of \( x' \) into \( v''(\lambda) \) results in:

\[ v''(\lambda) = \frac{1}{2} (1 - \lambda s)^{-3} \left[ 2\beta/v_2 \lambda (\beta-2c_1^2)(1-\lambda s)^{-1} \right] P(x,\lambda) \]

where

\[ P(x,\lambda) = P_1(x)\lambda^2 + P_2(x)\lambda + P_3(x) \]

and

\[ P_1(x) = \frac{4\beta/v \c_1 s^{2} - \frac{2}{s} s^2'}{s (s^2')^2 - 8\beta/v^2 \c_1^2 s^2} \leq 0 \]

\[ P_2(x) = \frac{-s(s^2')^2 + 4\beta^2/v^2 \frac{s}{s^2} - \frac{2}{s} s^2 - 4\beta/v \c_1^2 s^2}{s (s^2')^2 > 0} \]

\[ P_3(x) = -(s^2')^2 \leq 0 \]

We would like to remind readers that:

(i) \( 1 - \lambda s > 0 \)

(ii) \( (s^2')^2 - 8\beta/v^2 \c_1^2 s^2 > 0 \)

(iii) \( s^2' < 0 \)

(iv) \( A(x,\lambda) = 0. \)
Figure 4.4.2  $\varrho(x,\lambda)$ and $A(x,\lambda)$ for Fixed Value of $x$
We have shown in Section 4.3.2 that $\beta - 2c_1^2 > 0$. Therefore, $v'(\lambda)$ takes on the sign of $P(x, \lambda)$. For fixed values of $x$, $P(x, \lambda)$ is concave in $\lambda$ and $P(x, \lambda)$ has real roots in $\lambda$ also (by computing $P_2(x)^2 - 4P_1(x)P_3(x) > 0$). We plot $A(x, \lambda)$ and $P(x, \lambda)$ in Figure 4.4.2 for a fixed value of $x$.

If $A(x, \lambda)$ and $P(x, \lambda)$ intersect in the manner demonstrated in Figure 4.4.2, i.e., $\lambda^-_A$ and $\lambda^+_A$ are bracketed by $\lambda^-_p$ and $\lambda^+_p$, $v'(\lambda)$ is positive. This is because $P(x, \lambda^-_A)$ and $P(x, \lambda^+_A)$ are both positive. (We remind the reader that optimal $x$ satisfies $A(x, \lambda) = 0$, which corresponds to $\lambda^-_A$ and $\lambda^+_A$ for fixed $x$.) Unfortunately, we have been unable to prove this result.

4.5 Postscript

We would like to close this chapter with an example showing the behavior of $\overline{TR}(x)$ on a general network. Also, we will make some observations about the implications of minimizing $s^2(x)$ on a tree network.

4.5.1 Non-Convexity of $\overline{TR}(x)$ in a General Network

In locating an SQM over a tree network, we are able to exploit the tree structure and prove certain convex properties of $s^2(x)$ and $\overline{TR}(x)$. Such properties enable us to devise efficient search procedures to locate the SQM. We have proved convexity of $s^2(x)$ and $\overline{TR}(x)$ over a primary region of a link in a general network. Convexity, however, does not carry over to an entire link. This is seen intuitively because the set partition $A$ shrinks as one moves along a link. In a general network, the reason for $A$ shrinking across break-points is that we are keeping the reference nodes $a$ and $b$ fixed. When we prove our results on a tree network, we move along a path $i-j-k$ and the reference nodes change from $i$ and
We will first show that $s^2(x)$ and $\overline{TR}(x)$ are not convex on a link in a general network. An example is developed to show that multiple local minima of $\overline{TR}(x)$ exist on one link in a general network.

Consider two consecutive primary regions on link $(a,b)$. Let $x$ be the distance from the first reference node $a$. Let $c_i$ and $c_i'$ be the relevant $c_i$ parameters for the two consecutive primary regions, and $A'$, $B'$, $A''$, $B''$ be the respective node partitions $A$, $B$. Let $x$ be the break point between these two primary regions. Let $\ell$ be the length of $(a,b)$.

\begin{center}
\begin{tikzpicture}
\node (a) at (0,0) {$a$};
\node (b) at (4,0) {$b$};
\draw (a) -- (b);
\node (x) at (2,0) {$x$};
\node (c_i) at (1.2,0) {$c_i$};
\node (c_i') at (2.8,0) {$c_i'$};
\end{tikzpicture}
\end{center}

It is easy to see that

\begin{align*}
A' & \subset A, B' \subset B \\
A' \cup \sigma & = A \\
B'' & \subset B \\
A', B', \sigma & \text{ are mutually disjoint}
\end{align*}

and

\begin{align*}
C_4 & \leq C_4' \\
C_6 & \leq C_6' \\
C_1 & \leq C_1'
\end{align*}

where

$$\sigma = \{j| j \in N, \text{ and } d(j,a) + x = d(j,b) + \ell - x\}$$

Let $F(x)$ be a function defined on $(a,b)$, and
Figure 4.5.1 Multiple Local Optima of $\overline{TR}(x)$ on a Link.
\[ F(x)_l' = \lim_{\Delta \to 0} \frac{F(x) - F(x - \Delta)}{\Delta} = \text{left derivative of } F \text{ at } x \]

\[ F(x)_r' = \lim_{\Delta \to 0} \frac{F(x + \Delta) - F(x)}{\Delta} = \text{right derivative of } F \text{ at } x \]

When \( F \) is differentiable at \( x \), \( F(x)_l' = F(x)_r' \).

It is straightforward to show that

(i) \( \bar{s}^2(z)_l' \geq \bar{s}^2(z)_r' \)

and

(ii) \( \bar{TR}(x)_l' \geq \bar{TR}(x)_r' \)

by observing the changes of \( c_i \) across \( x \).

We now try to construct an example showing that \( \bar{TR}(x) \) can have multiple local minima on a link. Figure 4.5.1 shows the desired situation.

Instead of giving just a numerical example, we find it interesting to go through the construction of such an example.

Convexity of \( \bar{TR}(x) \) in each primary region allows us to evaluate four values of \( \bar{TR}(x)_l' \) to demonstrate this behavior: (i) \( \bar{TR}(x_1)_l < 0 \); (ii) \( \bar{TR}(x_2)_l > 0 \); (iii) \( \bar{TR}(x)_r < 0 \); and (iv) \( \bar{TR}(x_2)_l > 0 \).

Consider the following network:

There are two primary regions on link \((2,3)\) with break points at the midpoint of link \((2,3)\). We denote them by \(<0, 1/2>\) and \(<1/2, 1>\).
$A^\prime = \{1,2\} \quad A^\prime = \{2\}$

$B^\prime = \{3\}.$

$B^\prime = \{1,3\}$

Assume $\bar{w}_j = 1$ for $j = 1,2,3, \beta = 2, v = 1$. This implies $\bar{w} = 1, \bar{w}^2 = 1$.

We will leave the value $\varepsilon$ and the arrival rate $\lambda$ undetermined for the moment. First we compute all the relevant parameters.

$$
\begin{align*}
    c_1^\prime &= 0 \\
    c_2^\prime &= \varepsilon + \frac{1}{2} \\
    c_4^\prime &= \varepsilon - \frac{1}{2} \\
    c_6^\prime &= 0 \\
    c_1^\prime &= 2\varepsilon \\
    c_2^\prime &= 2\varepsilon + \frac{1}{2} \\
    c_4^\prime &= -2\varepsilon - \frac{1}{2} \\
    c_6^\prime &= -2\varepsilon \\
    c_5^\prime &= 4\varepsilon + \frac{1}{2} \\
    c_7^\prime &= 2\varepsilon + \frac{1}{2}
\end{align*}
$$

The value of $\lambda$ is such that $1 - \lambda \bar{s}(x) > 0$ for all $x \in [0,1]$, i.e., $x$ on link $(2,3)$. In primary region $<0, 1/2>$, $c_1^\prime = 0 = t(\cdot)^\prime = \bar{s}(\cdot)^\prime$. $\overline{TR}^\prime(\cdot) = \frac{\lambda}{2} \frac{s(\cdot)^\prime}{1 - \lambda \bar{s}(\cdot)^\prime}$. We want to show that $\overline{TR}(0)^\prime < 0$ and $\overline{TR}(1/2)^\prime > 0$. Since $s(0)^\prime = 2\beta^2/v^2 c_4^\prime + 2\beta/v c_6^\prime = 8(\varepsilon - 1/2) < 0$. $0 < \varepsilon < 1/2$, and $s(1/2)^\prime = 2\beta^2/v^2(1/2) + 2\beta^2/v^2 c_4^\prime + 2\beta/v c_6^\prime = 8\varepsilon > 0$. Therefore,

$$
\overline{TR}(0)^\prime < 0
$$

and

$$
\overline{TR}(1/2)^\prime > 0
$$

Over primary region $<1/2, 1>$ on link $(2,3),$

$$
\overline{TR}(\cdot)^\prime = \frac{\lambda}{2} (1 - \lambda \bar{s}(\cdot))^{-2} [(1 - \lambda \bar{s}(\cdot)) s(\cdot)^2 + \lambda \beta s^2(\cdot) \bar{t}(\cdot)] + \bar{t}(\cdot)^\prime.
$$
We want to show first that $\overline{\text{TR}}(1/2)_r < 0$, since $t(\cdot)' = \frac{1}{v} c_1'' = -2\varepsilon < 0$, and $s'^2(\frac{1}{2})_r = 2\beta^2/v^2(1/2) + 2\beta^2/v^2 c_4'' + 2\beta/v c_6'' = -24\varepsilon < 0$. Again, $1 - \lambda s(\cdot) > 0$ implies
\[ \therefore \overline{\text{TR}}(1/2)_r < 0. \]

Lastly, we have to evaluate $\overline{\text{TR}}(1)'_\lambda$ and show that for the appropriate choice of $\varepsilon$ and $\lambda$, $\overline{\text{TR}}(1)'_\lambda > 0$. Some of the quantities below depend on the choice of $\varepsilon$ and $\lambda$.

- $\overline{t}(1) = 1/2, \overline{s}(1) = \overline{w} + \beta \overline{t}(1) = 1 + 2 \cdot \frac{1}{2} = 2$
- $1 - \lambda \overline{s}(1) = 1 - 2\lambda$
- $\overline{t}(1)' = c_1'' = -2\varepsilon$
- $\overline{s}'(1) = 5$
- $\overline{s}'(1) = 4(1 - 6\varepsilon)$.

\[ \therefore \overline{\text{TR}}(1)'_\lambda = \frac{\lambda}{2} (1 - 2\lambda)^{-2}[4 - 24\varepsilon - 8\lambda + 28\lambda\varepsilon] \]

To ensure that $1 - \lambda \overline{s}(\cdot) > 0$ for all points on link (2,3), we impose an upper bound on $\lambda$: $\lambda < [2(1+\varepsilon)]^{-1}$. This is because $\max_x \overline{t}(x) = \varepsilon + 1/2, \overline{s}(x) = \overline{w} + \beta \overline{t}(x)$.

\[ \therefore 1 - \lambda \overline{s}(x) > 0 \]

implies $\lambda < \frac{1}{2(1+\varepsilon)} < \frac{1}{2}$, since $\varepsilon > 0$. 
Set \( \lambda = \frac{1}{2} (1-\Delta) < \frac{1}{2(1+\varepsilon)} \Rightarrow \Delta > \frac{\varepsilon}{1+\varepsilon} \) and \( \Delta < 1 \). Evaluating the left derivative of \( \overline{TR}(x) \) at \( x = 1 \), we have

\[
\overline{TR}(1)'_\lambda = \frac{1-\Delta}{4\Delta} [4\Delta - 10\varepsilon - 14\Delta \varepsilon].
\]

We want to choose \( \Delta \), and \( \varepsilon \) such that

(i) \( \overline{TR}(1)'_\lambda > 0 \)

(ii) \( \frac{\varepsilon}{1+\varepsilon} < \Delta < 1 \)

(iii) \( 0 < \varepsilon < 1/2 \)

Set \( \Delta = 5\varepsilon > \frac{\varepsilon}{1+\varepsilon} \), \( 0 < \varepsilon < 1/2 \) and \( \varepsilon \) small.

\[
\overline{TR}(1)'_\lambda = \frac{1}{10\varepsilon} [1 - 12\varepsilon + 15\varepsilon^2]
\]

for \( \varepsilon = \frac{1}{12} \), \( \overline{TR}(1)'_\lambda = \frac{15\varepsilon}{10} > 0 \).

Therefore, \( \Delta = \frac{5}{12} \), \( \lambda = \frac{7}{24} \), \( \varepsilon = \frac{1}{12} \) is one set of possible values of these parameters satisfying (i), (ii) and (iii) above. One finds the local minima are on link (2,3) at \( x = \frac{5}{12} \) and \( x = 0.9538 \).

The following example shows multiple local minima of \( \overline{TR}(x) \) can exist on a link in a general network (on \( x = 5/12 \) and \( x = 0.9538 \) on link (2,3)):
4.5.2 Minimization of $s^2(x)$ Over a Tree Network

We know that $s^2(x)$ is convex over any path in a tree network. There are two possibilities for the location of minimum $s^2(x)$ on a tree: (1) it is on a node; (2) it is in the interior of a link. When it occurs in the interior of a link, the necessary and sufficient condition for minimum $s^2(x)$ is $s^2(x)' = 0$. This condition reduces to

$$\sum_{j \in \mathcal{A}} h_j \left[ \beta / v \left( x + d(a,j) \right) + \bar{w}_j \right] = \sum_{j \in \mathcal{B}} h_j \left[ \beta / v \left( \ell - x + d(b,j) \right) + \bar{w}_j \right]$$

When the minimum of $s^2(x)$ is on link $(a,b)$, $\sum_{j \in \mathcal{A}} h_j \left[ \beta / v \left( x + d(a,j) \right) + \bar{w}_j \right]$ can be interpreted as workload generated from node set $\mathcal{A}$, where $\beta / v (x + d(a,j))$ is the round-trip time between the facility at $x$ and the incident at node $j$; $\bar{w}_j$ is the non-travel component of service time generated at node $j$. Likewise, $\sum_{j \in \mathcal{B}} h_j \left[ \beta / v \left( \ell - x + d(b,j) \right) + \bar{w}_j \right]$ can be seen as workload generated from set $\mathcal{B}$.

Therefore, minimization of $s^2(x)$, if it occurs on a link, can be viewed as equalization of workload between requests generated through node $a$ to $x$ and through node $b$ to $x$. When this happens on a node, it can be viewed as the "best" equalization (or splitting) of workload we can have.

Such an interpretation of workload equalization remains valid in a general network. However, the condition of workload equalization is only necessary, but not sufficient, for minimization of $s^2(x)$, because multiple local minima (of $s^2$) can exist on a general network. Another interesting interpretation is that, suppose we want to locate two units at the same location in a tree network and they are to operate independently to cover
two non-overlapping node sets, balancing workload also minimizes the second movement of service time.

It is interesting to contrast the optimality conditions of the Hakimi median problem and the minimization of $s^2(x)$. In the Hakimi problem, one tries to partition the node sets into A and B such that their respective total call rates are "more or less" balanced. It is called the median problem because the optimal location is the closest point to the 50th percentile of total arrival rate. Minimization of $s^2(x)$ partitions the node sets such that their respective workloads (travel plus non-travel time) are equalized. As will be seen in the next chapter, when we have continuous demand on the links of the network, the Hakimi median will be at the 50th percentile of total arrival rate if it is in the interior of a link.
Chapter 5

STOCHASTIC QUEUE MEDIAN ON A TREE NETWORK

WITH CONTINUOUS LINK DEMANDS

5.1 Preliminary

In this chapter, we generalize the results of Chapter 4 dealing with the location of the SQM on a tree network. In addition to nodal demands, we allow demands to arise on a link in a continuous manner, following a general probability density function. We believe that this is the first complete analysis to include link demand, in its most general form, in location theory on a network. Handler and Mirchandani [31] do formulate the p-median problem on a general network with discrete nodal and continuous link demands. On a tree network, they propose a solution procedure, which is a slight variant of the Goldman algorithm [22] for the one-median problem. However, no analysis is performed to characterize the behavior of the average travel time as a function of the location of the median. Minieka [57] implicitly considers link demands in a surrogate way. Specifically, he defines a general absolute median of a network to be a point on the network that minimizes the sum of (unweighted) distances from that point to the most distant point on each link.

Since researchers usually associate the Hakimi median with nodal location, we will term the point, which minimizes average travel time on a network with continuous link demands, the minisum location. It will become clear in the next section that the minisum location is, in general, not found at a nodal site. We will develop the necessary machinery characterizing the minisum location before studying the SQM problem. We recall from Chapter 4 that the minisum location is at the root of our
analysis.

In addition to the usual notations and problem definitions, we have
the following problem specifications.

- \( f_\ell \) = fraction of total demands originating on link \( \ell \)
  for \( \ell \in L \)

- \( \sum_{j \in N} \sum_{\ell \in L} f_\ell = 1 \)

- Demands on a link \( \ell \) follow a continuous proper density
  function \( f_\ell(y) \), \( y \in (0, \ell) \) where \( \ell = \) length of link \( \ell \).

  Without loss of generality, we do not allow impulses
  in the density function. This is because we can always
  place a fictitious node at the point where an impulse
  occurs. Also, \( \int f_\ell(y)dy = 1 \). By introducing \( f_\ell(y) \), we
  have imposed an orientation on link \( \ell \). We will discuss
  such concerns later.

- When the orientation of a link \( \ell \) is well defined, we have
  the following notations:

  \[
  F_\ell(x) = \int_0^x f_\ell(y)dy \\
  \overline{y}_\ell = \int_0^{\ell} f_\ell(y)dy.
  \]

  In the next section, we will first study the minisum location on a
general network in its entirety. The tree structure again, allows us to
prove certain convexity results on the functions \( \overline{t}(x) \), \( s^2(x) \) and \( \overline{TR}(x) \),
which play an important role in the development of an efficient algorithm
to search for the minisum location and the SQM. We will conclude this
chapter with a parametric analysis of the SQM trajectory as the total
arrival rate varies.
5.2 The Minisum Location on a Network with Continuous Link Demands

As an analytical tool to study the SQM on a network with continuous link demands, we need to characterize the minisum location together with the behavior of the average travel time. Since there is no result on this problem in the literature, we will engage in such a venture in this section. The first objective is to analyze the minisum problem on a general network. After recognizing the implications of a tree structure on the minisum problem, we develop convexity results that will aid us in locating the minisum site. The remainder of this chapter is concerned with the SQM problem on a tree network with continuous link demands.

5.2.1 General Undirected Network:

Consider a facility located at x on link (a,b), i.e., a point at a distance x from node a on link (a,b). A node j belongs to node partition A(x; a,b) if the shortest route from j to x passes node a; j belongs to node partition B(x; a,b) if the shortest route between j and x passes node b, specifically:

\[ A = \{ j \in N \mid d(j,a) + x < d(j,b) + \bar{m} - x \} \]

\[ B = N - A \]

where \( \bar{m} \) is the length of link (a,b).

We define breakpoints as points on link (a,b) at which the node partition changes. We observe that B gets bigger as one moves from node a to node b. A primary region is an interval on link (a,b), in which there is no breakpoint. When we have link demands, we also have to
partition the link set into AL and BL. In the presence of a cycle in a general network, we may have to split up a link in the partitioning of AL and BL. This complicates the analysis enormously.

Again we define the primary region exactly as in Section 3.2.1. In a primary region, we do not have changes in the node partition. However, link partitions can change continuously. We classify each link into one of the following three types: (i) both end nodes belong to A; (ii) both end nodes belong to B; (iii) one end node is an A node, the other a B node. We will consider the classification of link \((i,j)\). Link \((i,j)\) has a special orientation, namely from node \(i\) to node \(j\). When link \((i,j)\) is an AB link, without loss of generality, we always assume \(i\) to be an A node and \(j\) a B node. We now consider the following cases:

\[
\begin{array}{c}
\text{i} \\
\hline
\text{Z} \\
\hline
\text{j}
\end{array}
\]

\(\text{Z}_{\overline{x}}\)

\(\overline{x}\)

(i) \(i \in A\) and \(j \in A\), and the length of \((i,j)\) is \(\overline{x}\). We observe that there exists a point \(Z_{\overline{x}}\) on \((i,j)\) such that for \(0 \leq y \leq Z_{\overline{x}}\), the shortest route from \(y\) to node \(a\) is via node \(i\); and for \(Z_{\overline{x}} < y \leq \overline{x}\), the shortest route from \(y\) to node \(a\) is via node \(j\). Specifically,

\[Z_{\overline{x}} + d(i,a) + x = \overline{x} - Z_{\overline{x}} + d(j,a) + x\]

or

\[Z_{\overline{x}} = \frac{[d(j,a) - d(i,a) + \overline{x}]}{2}\]

Note that \(0 \leq Z_{\overline{x}} \leq \overline{x}\) because \(d(j,a) \leq d(i,a) + \overline{x}\) and \(d(i,a) \leq d(j,a) + \overline{x}\).
We notice that the value $Z_\lambda$ remains unchanged in the corresponding primary region of link $(a,b)$. When one moves across breakpoints, $i$ may turn into a B node, $j$ may turn into a B node, or both may turn into B nodes.

(ii) $i \in B$ and $j \in B$.

Exactly the same analysis applies here, except that $Z_\lambda$ now takes on the following value

$$Z_\lambda = \frac{1}{2} [d(j,b) - d(i,b) + \bar{\lambda}]$$

A moment's thought will convince the reader that nodes $i$ and $j$ remain "B" nodes as the facility moves away from node $a$ on link $(a,b)$.

(iii) $i \in A$ and $j \in B$.

Again, we can identify a point $Z_\lambda$ on link $(i,j)$.

However, this separation point $Z_\lambda$ moves as $x$ moves even within the same primary region on link $(a,b)$. $Z_\lambda$ satisfies the following relationship:

$$Z_\lambda + d(i,a) + x = \bar{\lambda} - Z_\lambda + d(j,b) + m - x$$
or

$$Z_\lambda = \frac{1}{2} [m + \bar{\lambda} + d(j,b) - d(i,a) - 2x]$$

Also, $0 \leq Z_\lambda \leq \bar{\lambda}$ because

$$d(j,b) + m - x < d(i,a) + \bar{\lambda} + x \quad (j \in B)$$
$$d(i,a) + x < \bar{\lambda} + d(j,b) + m - x \quad (i \in A)$$

We see that the point $Z_\lambda$ moves continuously as $x$ moves within a primary region on link $(a,b)$. In fact, $Z_\lambda$ decreases as $x$ increases. Therefore, more
demands on link \((i,j)\) will turn into BL type demands continuously as would be expected.

From the analysis and classification above, we define type (i) links as AL links, type (ii) links as BL links, and type (iii) links as ABL links. We also observe that BL links remain BL links as the facility moves from node \(a\) to node \(b\) on link \((a,b)\). Within a primary region, we can evaluate the contribution of service demand to the total expected travel distance as follows:

(1) Nodal demands:

\[
(\Sigma h_j - \Sigma h_j)x + \Sigma h_jd(a,j) + \Sigma h_j[d(b,j) + m] = c_1x + c_2
\]

where \(m\) is the length of link \((a,b)\) on which the facility is being considered at point \(x\).

(2) Link demands:

(a) for all \(\ell \neq (i,j) \in AL, \ell \neq (a,b)\)

\[
L_\ell = f_\ell \left[ \int_0^{Z_\ell} [y + d(i,a) + x]f_\ell(y)dy + \int_{Z_\ell}^{\bar{\ell}} [-y + d(j,a) + x]f_\ell(y)dy \right]
\]

where \(Z_\ell = \frac{1}{2} [d(j,a) - d(i,a) + \bar{\ell}]\).

We simplify \(L_\ell\) as follows:

\[
L_\ell = f_\ell \left[ x + d(j,a) + \bar{\ell} + [d(i,a) - \bar{\ell} - d(j,a)]f_\ell(Z_\ell) - \bar{y}_\ell + 2\bar{y}(Z_\ell) \right]
\]

where \(\bar{y}_\ell(Z_\ell) = \int_0^{Z_\ell} yf_\ell(y)dy\)

and

\[
\bar{y}_\ell = \int_0^{Z_\ell} yf_\ell(y)dy
\]

Note that \(Z_\ell\) does not depend on \(x\), the only term
in $L_{\ell}$ dependent on $x$ is $f_{\ell} x$.

(b) $\ell = (i,j) \in BL. \ell \neq (a,b)$.

$$L_{\ell} = \varepsilon \int_0^x \int_{\ell} [y + d(i,b) + m - x]f_{\ell}(y)dy \int_{\ell} [x - y + d(j,b) + m - x]f_{\ell}(y)dy$$

where $Z_{\ell} = \frac{1}{2} [d(j,b) - d(i,b) + \ell]$

$$L_{\ell} = -f_{\ell} x + f_{\ell} (m + \ell + d(j,b) + [d(i,b) - \ell - d(j,b)]F_{\ell}(Z_{\ell}) - y_{\ell} + 2y_{\ell}(Z_{\ell})]$$

Again, $L_{\ell}$ depends on $x$ only through $-f_{\ell} x$.

(c) $\ell = (i,j) \in ABL. \ell \neq (a,b)$.

$$L_{\ell} = \varepsilon \int_0^x \int_{\ell} [y + d(i,a) + x]f_{\ell}(y)dy \int_{\ell} [\ell - y + d(j,b) + m - x]f_{\ell}(y)dy$$

where $Z_{\ell} = \frac{1}{2} [m + \ell + d(j,b) - d(i,a)] - x$

$$L_{\ell} = f_{\ell} ([1 - F_{\ell}(Z_{\ell})](\ell + m + d(j,b) - x) + (d(i,a) + x)F_{\ell}(Z_{\ell}) + 2y_{\ell}(Z_{\ell})] - f_{\ell} y_{\ell}$$

There are many $x$-dependent terms in $L_{\ell}$.

(d) $\ell = (a,b) = m$. We assume triangular inequality holds.

$$L_m = \varepsilon \int_0^x (x - y)f_m(y)dy \int_x^m (y - x)f_m(y)dy$$

$$= f_m [xF_m(x) - x(1 - F_m(x)) + y_{\ell} - 2y_{\ell}(x)]$$

The objective, as in the traditional Hakimi median problem, is to minimize the total expected travel time $\bar{t}(x)$ on the network.

$$\bar{t}(x) = c_1 x + c_2 + \sum_{\ell \in AL} L_{\ell} + \sum_{\ell \in BL} L_{\ell} + \sum_{\ell \in ABL} L_{\ell} + L_m$$

$$\ell \neq m \ell \neq m \ell \neq m$$
**Definition:** A point \( y \) on link \((a,b) = m\) is called the minisum location on a general network with continuous link demand if \( t(y) \leq t(x) \) for all \( x \) on \( G(N,L) \).

However, it is not possible to determine the curvature of \( t(x) \) even in a primary region. The intuitive reason is: even within a primary region where no change of nodal partition takes place, the AL portion of an ABL link is "defecting" to become a BL portion - continuously. This movement induces concavity in \( L_\lambda \). Our definition of a primary region allows infinitesimal shifting of the probability mass from "A" demands to "B" demands. This can be seen in the case of the nodal-demand-only network, where \( t(x) \) is linear in a primary region and concave across a breakpoint. The contribution to \( t(x) \) from link \((a,b)\), however, is convex in \( x \), and therefore the curvature of \( t(x) \) is not determined. The implication of this is that there may exist multiple minima of \( t(x) \) even in a primary region. This makes the search procedure more difficult. One may have to solve for all the zeroes of \( t(x) \) in a primary region and then over all the primary regions. This involves the identification of breakpoints (as discussed in Berman Larson and Chiu [3]) and searches over all primary regions.

We conclude this section by stating a negative result:

**Theorem 5.2.1:** The curvature of \( t(x) \) is undetermined over a primary region in a general undirected network with continuous link demands.

**Proof:** We will consider each component of \( t(x) \):

1. nodal contribution \( c_1 x + c_2 \) linear;
2. link contribution, \( \ell \in AL \) or BL; \( L_\lambda \) is linear in \( x \);
(3) link contribution $\lambda \in ABL$, recalling that $Z_{\lambda} = \frac{1}{2} (\bar{m} + \bar{\lambda} + d(j,b) - d(i,a)) - x$.

$$\frac{dL_{\lambda}}{dx} = \left[2F_{\lambda}(Z_{\lambda}) - 1\right]f_{\lambda}$$

$$\frac{d^2L_{\lambda}}{dx^2} = f_{\lambda}(-2f_{\lambda}(Z_{\lambda})) \leq 0 \implies L \text{ concave; }$$

(4) link contribution $\lambda = m = (a,b)$.

$$\frac{dL_{m}}{dx} = (2F_{m}(x) - 1)f_{m}$$

$$\frac{d^2L_{m}}{dx^2} = 2f_{m}f_{m}(x) \geq 0 \implies L_{m} \text{ convex.}$$

Therefore, we cannot determine the curvature of $\bar{t}(x)$ over a primary region. However, it is likely that $\bar{t}(x)$ is concave because the contribution of the second derivative comes from, hopefully, many ABL links and only one $(a,b) = m$ link which is convex.

$$\frac{d^2\bar{t}(x)}{dx^2} = 2\left[-\sum_{\lambda \in ABL} f_{\lambda}f_{\lambda}(Z_{\lambda}) + f_{m}f_{m}(x)\right]$$

where

$$Z_{\lambda} = \frac{1}{2} (\bar{m} + \bar{\lambda} + d(j,b) - d(i,a)) - x.$$ 

Except for abnormally large values of $f_{m}$ and $f_{m}(x)$, we can heuristically check only the breakpoints for the optimal minimum location. We also observe that in a tree network, there is no ABL link; therefore, $\bar{t}(x)$ is convex over the entire link.
Before we specialize the minisum problem in the tree network, we pause for some numerical examples illustrating the type of link classifications discussed above. Figure 5.2.1 shows link (a,b) serving as a bridge between two sub-networks, A and B. All links/nodes in A(B) are A(B) links/nodes. There are no ABL links for facilities located on link (a,b). Figure 5.2.2 shows link (i,j) as an ABL link when the facility is located at x on link (a,b). The length of each link is indicated next to the edge. The point $Z_L$ separates link (i,j) into two segments: for $y$ on (i,j), $0 \leq y \leq Z_L$ demands in this segment will be served by a facility at x on (a,b) via node i and node a; for $y$ on (i,j), $Z_L \leq y \leq 2$ demands are served via node j, then node b to point x. We calculate $Z_L = 2-x$. The implication is that more demands on link (a,j) become "B" demands as x moves from node a to node b.

Figure 5.2.3 shows link (i,j) as an AL link. However, part of link (i,j) has its shortest route to node a via node i and the rest through node a via node j. Here, $Z_L = 2.5$ on link (i,j) from node i. For $y$ on link (i,j) and $0 \leq y \leq 2.5$, the shortest route to facility x on (a,b) is via node i; for $2.5 \leq y \leq 4$, the shortest route is via node j.

We will now specialize our effort on tree networks. The first objective is to characterize the minisum location and the associated average travel time. An efficient algorithm, similar to that of Mirchandani and Handler [31], is developed to locate the minisum location on a tree network. We will prove parallel convexity results of the functions $\frac{s^2}{2}$ and $\overline{TR}$. This chapter ends with parametric analysis on the SQM location as arrival rate varies.
Figure 5.2.1  No ABL Links associated With Locations on (a,b).

Figure 5.2.2  Segmentation of Link (i,j), an ABL Link.

Figure 5.2.3  Segmentation of Link (i,j), an AL Link.
5.2.2 On a Tree Network:

A general expression for the expected travel time, when the facility
is located at \( x \) on link \( (a,b) \equiv m \), is

\[
\bar{t}(x) = \left[ \sum_{j \in N} h_j d(j, x) + \sum_{\lambda \in L} \int_{y \in \lambda} d(y, x) f_{\lambda}(y) dy + \int_{y \in m} |y-x| f_m(y) dy \right] / v
\]

We are considering the location on link \( (a,b) = m \). For a tree net-
work, the node partitioning \( A \) and \( B \) depends only on the link on which
the facility is located, and not on the exact location \( x \) on the link.
We now define the node and link partitioning specializing on a tree:

\[
A \equiv \{ j \mid j \in N \text{ and } d(j, a) < d(y, b) \}
\]

\[
B \equiv N - A
\]

\[
AL \equiv \{ \ell \mid \ell \in L, \ell \neq m, \text{ and for all } y \in \ell, d(y, a) < d(y, b) \}
\]

\[
BL \equiv L - AL
\]

\[
BL^- \equiv BL - \{ m \}
\]

In words, the node in set \( A \) communicates with points on \( (a,b) \) via node \( a \).
The set of nodes \( B \) communicates with points on \( (a,b) \) via node \( b \). \( AL \) and
\( BL \) are the corresponding link sets. We have included link \( (a,b) \) in \( BL \).

As mentioned earlier, we have to define an orientation for each
link when we integrate its continuous demands along its length.
Definition: \( \lambda(a) \) = approaching \(((a,b))\) node of link \( \lambda \). \( \lambda(r) \) = receding (from \((a,b))\) node of link \( \lambda \); i.e., \( d(\lambda(r),a) = d(\lambda(a),a) + \lambda \). Note that in a tree network, \( d(j,b) = d(j,a) + m \) for \( j \in A \), and \( d(j,a) = d(j,b) + m \) for \( j \in B \). We also define \( m(a) = a \), \( m(r) = b \) as a convention.

When we integrate the link demand density, we always move from node \( \lambda(a) \) to node \( \lambda(r) \). We will now express \( \overline{t}(x) \) in terms of the node and link partitions:

\[
\overline{t}(x) = \frac{1}{V} \left\{ \left( \sum_{j \in A} h_j - \sum_{j \in B} h_j \right) + \left( \sum_{\lambda \in AL} f_{\lambda} - \sum_{\lambda \in BL} f_{\lambda} \right) \right\} x + 2f_m x F_m(x) - \\
- 2f_m \int_{0}^{x} yf_m(y)dy + \sum_{j \in N} h_j d(j,a) + \sum_{\lambda \in L} f_{\lambda} \overline{y}_{\lambda} + \sum_{\lambda \in L} f_{\lambda} d(\lambda(a),a) \}
\]

One can write:

\[
\overline{t}(x) = \overline{t}' + \overline{t}_0 + \frac{1}{V} \left( 2f_m x F_m(x) - 2f_m \int_{0}^{x} yf_m(y)dy \right)
\]

where

\[
\overline{t}_0 = \overline{t}(x) \bigg|_{x=0}
\]

\[
\overline{t}'_0 = \frac{dt(x)}{dx} \bigg|_{x=0}
\]

We want to emphasize that \( A, B, AL \) and \( BL \) remain the same for all \( x \) on \((a,b) = m\) in a tree network.

Lemma 5.2.1: \( \overline{t}(x) \) is convex for all \( x \) on a link in a tree network.

Proof: \( \frac{d^2 \overline{t}(x)}{dx^2} = \frac{2}{V} f_m f_m(x) \geq 0 \). \( \therefore \overline{t}(x) \) is convex.
**Lemma 5.2.2:** $\bar{t}(x)$ is convex along any path in a tree, where $x$ is the distance measured from one end of the path.

**Proof:** Since $\bar{t}(x)$ is convex on a link, we need only concern ourselves with the behavior of $\bar{t}(x)$ across a node. Proof techniques are the same as in the case of nodal-demand-only. We now have to consider the changes in $A$, $B$ as well as $AL$ and $BL$. The definitions of in and out derivatives at a node along an implied path are the same as in Section 4.2.3.

Consider a path $i - j - k$. We associate $A'$, $B'$, $AL'$, $BL'$ with link $(i,j)$ and $A''$, $B''$, $AL''$, $BL''$ with link $(j,k)$. The lemma is proved if $\bar{t}(j)_\text{in} \leq \bar{t}(j)_\text{out}$ along $i - j - k$. Let

$$C_1 = \sum_{j \in A} h_j - \sum_{j \in B} h_j$$

$$CL_1 = \sum_{l \in AL} f_l - \sum_{l \in BL} f_l$$

$$\bar{C}_1 = C_1 + CL_1$$

and

$$\bar{x}' = \text{length of link } (i,j)$$

$$\bar{x}'' = \text{length of link } (j,k)$$

We will refer links $(i,j)$ and $(j,k)$ as links $\bar{x}'$ and $\bar{x}''$.

**Definition:** $\Delta(j; i,k) = \text{set of nodes connected to } j \text{ (including node } j\text{)}$ when we remove links $(i,j)$ and $(j,k)$. $\Delta L(j; i,k) = \text{set of links connected to node } j \text{ when we remove links } (i,j) \text{ and } (j,k)$.

When there is no ambiguity, we simply refer to $\Delta$ and $\Delta L$. It is easy to show that


\[ A'' = A' \cup \Delta, \quad B' = B'' \cup \Delta \]

\[ AL'' = AL' \cup \Delta L \cup \{ \ell' \} \]

\[ BL' = BL'' \cup \Delta L \cup \{ \ell' \} \]

Note that \( t'(x) = \frac{1}{v} [C_1 + CL_1 + 2f F_m(x)], \) when \( x \) is on link \( m. \) Along \( i - j - k, \)

\[ t'(j)_{in} = \frac{1}{v} [C_1' + CL_1' + 2f \lambda'] \]

because on link \( (i,j), \) \( F_{\lambda},(x)_{x=\lambda'} = 1, \)

\[ t'(j)_{out} = \frac{1}{v} [C_1'' + CL_1'' + 2f \lambda'' F_{\lambda''}(x)_{x=0}] = \frac{1}{v} (C_1'' + CL_1''). \]

We can show that \( C_1'' = C_1' + 2 \sum p \in \Delta p, \quad CL_1'' = CL_1' + 2 \sum l \in \Delta L \lambda f \lambda \)

\[ \therefore t'(j)_{out} = \frac{1}{v} [C_1' + CL_1' + 2f \lambda'] + \frac{1}{v} [2 \sum p \in \Delta p + 2 \sum l \in \Delta L \lambda f \lambda] \geq t'(j)_{in} \]

Since \( \bar{t}(x) \) is convex on any path of a tree, lemma 4.2.8 applies to \( \bar{t}(x). \)

Suppose \( i, j \) are two adjacent nodes on a tree network, and \( \bar{t}(i) < \bar{t}(j). \)

Then \( \bar{t}(x) > \bar{t}(j) \) for all \( x \) on \( (j,k), \) where \( k \neq i. \)

5.2.3 An Efficient Algorithm to Locate the Minisum Point on a Tree with Continuous Link Demands:

We will make several observations before presenting an efficient algorithm to locate this "extended" Hakimi median.
We can lump all mass $f_\ell$ (of link $\ell$) at a distance $\bar{y}_\ell$ on link $\ell$ from node $\lambda(a)$ (the approaching node of $\lambda$); $\bar{t}(x)$ becomes

$$\bar{t}(x) = \frac{1}{v} \left\{ \left( \sum_{j \in A^+} h_j - \sum_{j \in B^+} h_j \right) x + 2f_{m(x)} x + 2f_m \int_0^x yf_m(y)dy + \sum_{j \in N^+} h_d(j,a) \right\}$$

where $A^+$, $B^+$ and $N^+$ include all the fictitious nodes with mass $f_\ell$ on link $\ell$ at a distance $\bar{y}_\ell$ from node $\lambda(a)$. We have, then, essentially a discrete-nodal-demand version of the classic Hakimi median problem except for two non-linear terms. Mirchandani and Handler make a similar observation in [31].

Another more useful observation is to lump all mass $f_\ell$ on node $\lambda(a)$ for each $\ell \neq m$; $\bar{t}(x)$ becomes

$$\bar{t}(x) = \frac{1}{v} \left\{ \left( \sum_{j \in A^+} h_j - \sum_{j \in B^+} h_j \right) x - f_{m} x + 2f_{m(x)} x + 2f_m \int_0^x yf_m(y)dy + \text{constant terms} \right\}$$

where $h_j^+$ includes all the link masses $f_\ell$ where $j = \lambda(a)$ (except for link $(a,b) = m$).

If the minimum of $\bar{t}(x)$ occurs on the interior of link $(a,b)$, $\bar{t}(x)' = 0$ or

$$\sum_{j \in A^+} h_j^+ - \sum_{j \in B^+} h_j^- - f_m + 2f_{m(x)} = 0$$
or
\[ \sum_{j \in A} h_j^+ + f_m(x) = \sum_{j \in B} h_j^+ + f_m(1 - F_m(x)) = \frac{1}{2} \]
since
\[ \sum_{j \in A} h_j^+ + \sum_{j \in B} h_j^+ + f_m = 1 \]
and
\[ 0 \leq F_m(x) \leq 1 \quad \text{for} \ x \in (0, m). \]

The point \( x \) is truly a median in the probabilistic sense. Intuitively, if we move the facility away from \( x \), we are sacrificing more than half of the demands into making a longer trip to the facility. Thus, we will increase the expected travel time. Handler and Mirchandani make the same observation in [31].

- The minisum location is at node \( j \) if the following is true:

  along any path through node \( j \), \( \bar{t}(j)_{\text{in}} \leq 0 \) and \( t(j)_{\text{out}} \geq 0 \).

  This is because \( \bar{t} \) is convex along any path on a tree network. We now explore the implications of this observation and develop a meaningful algorithmic procedure to detect such situations. Consider the situation in Figure 5.2.4.; node \( j \) is our point of focus. There are \( k \) links incident on node \( j \); the adjacent nodes are \( j_1, j_2, \ldots, j_k \). The aggregated weights of branch \( j_i \) and link \((j_i, j)\) are denoted by \( W(J_i) \). Note that \( \sum_{i=1}^{k} W(J_i) + h_j = 1 \). We investigate the implications of \( t(j)_{\text{in}} \geq 0 \) and \( t(j)_{\text{out}} \geq 0 \) as we move along a path through node \( j \), say \( j_p \rightarrow j \rightarrow j_q \).
(i) $\overline{t(j)}_{in} \leq 0$ implies $W(J_p) \leq \frac{1}{2}$

(ii) $\overline{t(j)}_{out} \geq 0$ implies $h_j + \sum_{i \neq q} W(J_q) \geq \frac{1}{2}$.

Before we make these inequalities operational, we make the following definition and observation:

**Definition:** The degree of a node $j = d(j) =$ number of links incident on node $j$.

**Definition:** An end of a tree is a node with degree one.

We know that:

(a) a tree has at least two ends;

(b) deleting an end and its lone link in a tree network will result in a new tree.

Now consider the following operation: start from an end node in branch $j_1$ and aggregate its weight together with its link weight to its adjacent node; delete this end node and its lone link. Suppose we continue this operation up to node $j$ and update the nodal weight of node $j$ (after the gradual deletion of branch $j_1$ and link $(j_1, j)$) by $h_j + W(J_1)$ and test whether this aggregated weight exceeds $1/2$. Repeating this procedure with another branch $j_1$, we will encounter a situation where the updated $h_j$ exceeds $1/2$ (as guaranteed by inequality (ii) above). We know that node $j$ is the desired minisum location.

The following procedure locates the minisum point of a tree network:

1. Locate an end of a tree, say $i$, with its connecting link $(i, j) = \ell$;
Figure 5.2.4 Minisum Location at Node j.

Figure 5.2.5 Graph of $F(x)$ and $y$. 
(2) Check \( h_i \geq 1/2 \). If yes, node \( i \) is the minimum location; if no, go to (3).

(3) Check \( h_i + f \xi \geq 1/2 \). If yes, locate the minimum point on link \((i,j) = \xi \); if no, update \( h_j \) to \( h_i + h_j \).

(4) Check \( h_j \geq 1/2 \). If yes, node \( j \) is the minimum point; if no, delete node \( i \) and its link \((i,j) \). Go to (1).

Since there are only a finite number of links and nodes, the above algorithm converges in a finite number of steps.

We still have to specify the operation in step (3), namely, to locate the minimum point on link \((i,j) = \xi \). We have to find a point \( y \), such that

\[
\sum_{k \in A} h_k + f \xi F_k(y) = 1/2.
\]

Note that, according to our construct, set \( A \) consists of the single node \( i \) whose node weight \( h_i \) has been updated to include all the node and link weights of the subtree \( T_i \), \( i \in T_i \), after the removal of link \((i,j) \).

Therefore, we wish to find such that

\[
f \xi F_k(y) = 1/2 - h_i.
\]

or

\[
y = F^{-1}_\xi[1/f \xi(1/2 - h_i)].
\]

Figure 5.2.5 shows the graph \( F_\xi(x) \). Note that, according to our algorithm,

\[
h_i < 1/2, \ h_i + f \xi \geq 1/2
\]

which imply \( 1/f \xi(1/2 - h_i) > 0 \) and \( 1/f \xi(1/2 - h_i) < 1 \). y exists because \( F_\xi(x) \)
Find an end node, i, and its associated link \((i,j) = \ell\)

\[ h_i \geq \frac{1}{2} \]  

- YES: Node i is the minisum location

\[ h_i + f_\ell \geq \frac{1}{2} \]  

- YES: The minisum location is at \(x\) on \((i,j)\) with:
  \[ x = F_\ell \left( \frac{1}{f_\ell} \left( \frac{1}{2} - h_i \right) \right) \]

- NO: Delete node i and its link \((i,j)\)

Update \(h_j\) with:
\[ h_j + f_\ell + h_i \]

\[ h_j \geq \frac{1}{2} \]  

- YES: Node j is the minisum location

STOP

Figure 5.2.6 Flowchart: Minisum Location On a Tree Network with Continuous Link Demands.
being a cumulative probability distribution function, is monotonically non-decreasing. Also, there is no discontinuity in $F_x(x)$ since no impulses are allowed. Figure 5.2.6 shows the flow chart of our algorithm.

5.2.4 A Numerical Example

We will illustrate our minisum location algorithm with a numerical example, as shown in Figure 5.2.7. Nodal weights $h_j$ are shown next to the nodes. Link lengths and link weights are shown next to the links as length/weight. The algorithm iterates by randomly selecting an end of a tree (one may streamline the algorithm by selecting an end with the largest weight).

1. Select end node 7, $h_7 = 0.019 < 1/2$
   
   link (7,10) with weight 0.04
   
   $0.019 + 0.04 = 0.059 < 1/2$
   
   $h_7 + h_{10} + f(7,10) = 0.019 + 0.002 + 0.04 = 0.061 < 1/2$
   
   update $h_{10} = 0.061$ Figure 5.2.7(b)

2. Select end node 3, $h_3 = 0.298 < 1/2$
   
   link (3,2) with weight 0.03
   
   $0.298 + 0.03 = 0.301 < 1/2$
   
   $h_3 + h_2 + f(3,2) = 0.298 + 0.12 + 0.03 = 0.448$
   
   update $h_2 = 0.448$ Figure 5.2.7(c)

3. Select end node 2, $h_2 = 0.448 < 1/2$
   
   link (2,1) with weight 0.06
   
   $0.448 + 0.06 = 0.508 > 1/2$

Minisum location is on link (2,1)
Figure 5.2.7 Example: Minisum Location on a Tree Network with Continuous Link Demands.
The probability density function for demands on link (2,1) is uniform: \( f(y) = \frac{1}{2}, \ 0 \leq y \leq 2 \). Therefore the minimum location is on link (2,1) at a distance \( x \) from node 2, where \( x \) satisfies:

\[
0.06 \int_{0}^{x} \frac{1}{2} \, dy + 0.448 = 0.5
\]

\[
0.03x + 0.448 = 0.5
\]

\[
x = 1.73
\]

Therefore the minimum location is on link (2,1) at a distance 1.73 from node 2.

Before moving on to the next topic, we would like to point out that no restriction has been imposed on the form of the link density functions except for continuity. This is expected since we can impose nodal demands by successively approximating the demand density using fictitious nodes. Therefore, the forms of the density functions should not come into the analysis. Finally, the median does not necessarily coincide with a node.

5.3 Properties of \( s^2 \) and TR on a Tree Network

Having characterized the minimum (average travel time) function over a tree network, we proceed now to examine the behavior of \( s^2 \) and TR. The minimum location problem is appropriate under the assumption of infinite server capacity. We now go back to the world of queueing and its effect on the location of facilities.

Techniques of proof are similar to those in Chapter 4. The non-polynomial terms (in \( x \)) introduced by link demands, however, do produce
complications in the analysis. All the convexity results do carry over in this case.

All notations and conventions are the same as in the case of the minisum location problem studied in Section 5.2. In addition, we have a non-travel related component of the service time, \( \tilde{w}_\ell \) for each link \( \ell \in L \) and \( \tilde{w}_j \) for each node \( j \in N \). We assume that \( \tilde{w}_\ell \) and \( \tilde{w}_j \) both have finite means and second moments, and are stochastically independent of travel time and the location of the facility. \( \tilde{w}_\ell \) has the same distribution at each point on the same link \( \ell \). A constant travel speed, \( v \), is observed on all links and a \( \beta \) factor is assumed to account for round trip travel time. We will first study \( \overline{s^2}(x) \) and then \( \overline{TR}(x) \).

5.3.1 Properties of \( \overline{s^2}(x) \) on a Tree Network:

For a facility located at \( x \) on link \( (a,b) = m \), we can write \( \overline{s^2}(x) \) as:

\[
\overline{s^2}(x) = \sum_{j \in A} h_j \left( \frac{w_j}{\beta + \sqrt{v^2 d(j,x)}} \right)^2 + \sum_{\ell \in L} f_{\ell} \int_{y \in \ell, \ell \neq m} \left( \frac{w_\ell}{\beta + \sqrt{v^2 d(y,x)}} \right)^2 f_{\ell}(y) dy
\]

\[
+ f_m \int_{y \in m} \left( \frac{w_m}{\beta + \sqrt{v^2 |y-x|}} \right)^2 f_m(y) dy
\]

With the same nodal and link partitions \( A, B, AL \) and \( BL \), we can write \( \overline{s^2}(x) \) as follows (note that \( m = (a,b) \in BL \)):

\[
\overline{s^2}(x) = \frac{\beta^2}{v^2} + \left[ \frac{2\beta^2}{v^2} C_4 + \frac{2\beta}{v} C_6 \right] + \frac{2\beta^2}{v^2} C_{L_4} + \frac{2\beta}{v} C_{L_6} + \frac{2\beta^2}{v^2} C_{L_8} x
\]

\[
+ \frac{4\beta}{v} \int_{m} \tilde{w}_m F_m(x) x - \frac{4\beta}{v} \int_{m} \tilde{w}_m \int_{0}^{x} yf_m(y) dy + K
\]
where

\[ C_4 = \sum_{j \in A} h_j d(j, a) - \sum_{j \in B} h_j d(j, a) \]

\[ C_6 = \sum_{j \in A} w_j - \sum_{j \in A} w_j \]

\[ CL_4 = \sum_{\lambda \in AL} f_\lambda d(\lambda(a), a) - \sum_{\lambda \in BL} f_\lambda d(\lambda(a), a) \]

\[ CL_6 = \sum_{\lambda \in AL} f_\lambda w_\lambda - \sum_{\lambda \in BL} f_\lambda w_\lambda \]

\[ CL_8 = \sum_{\lambda \in AL} f_\lambda y_\lambda - \sum_{\lambda \in BL} f_\lambda y_\lambda \]

\[ K = \text{constant} = \beta^2 \mu^2 \sum_{j \in N} h_j^2 [d(j, a)]^2 + 2\beta \mu \sum_{j \in N} h_j w_j d(j, a) + \sum_{j \in N} w_j^2 \]

\[ + \beta^2 \mu^2 \sum_{\lambda \in L} f_\lambda^2 [d(\lambda(a), a)]^2 + 2\beta \mu \sum_{\lambda \in L} f_\lambda w_\lambda d(\lambda(a), a) + \sum_{\lambda \in L} w_\lambda^2 \]

\[ + \beta^2 \mu^2 \sum_{\lambda \in L} f_\lambda^2 y_\lambda^2 + 2\beta \mu \sum_{\lambda \in L} f_\lambda w_\lambda y_\lambda + 2\beta^2 \mu^2 \sum_{\lambda \in L} f_\lambda d(\lambda(a), a) y_\lambda \]

We remind our readers of the convention that: (i) \( m(a) = a \), \( m(r) = b \), therefore \( d(m(a), a) = 0 \); and (ii) all integration along a link is performed from \( \lambda(a) \) to \( \lambda(r) \).

The following observations allow us to write \( s^2(x) \) in a more convenient form:

\[ \cdot [F_m(x) - \int_0^x y f_m(y) dy] \bigg|_{x=0} = 0 \]

\[ \cdot \frac{d}{dx} [F_m(x) - \int_0^x y f_m(y) dy] \bigg|_{x=0} = F_m(x) \bigg|_{x=0} = 0 \]

\[ \cdot \text{Let } G_0 \text{ represent the value of a function } G \text{ evaluated at zero.} \]
Therefore:

\[
\bar{s}^2(x) = \beta^2/v^2 x^2 + \bar{s}_0^2 x + \bar{s}_0^2 + 4\beta/v \int_0^x yf_m(y)dy
\]

The first result concerns convexity of \(\bar{s}^2(x)\) on a link.

**Lemma 5.3.1:** \(\bar{s}^2(x)\) is convex over a link on a tree network.

**Proof:**

\[
\frac{d^2\bar{s}^2(x)}{dx^2} = \frac{2\beta^2}{v^2} + 4\beta/v \int f_m f_m(x) > 0.
\]

Before we investigate the behavior of \(\bar{s}^2(x)\) over a path, we will discuss a subtle point concerning the direction of integration along a link. This is of concern to us because the direction changes on certain links when one moves across a node.

Consider a link \(\ell = (i,j)\) with length \(\ell\). We have defined \(y_\ell = \int_{y=0}^{y=\ell} yf_\ell(y)dy\), where we integrate from \(i = \ell(a)(y = 0)\) to \(j = \ell(r)(y = \ell)\). We want to point out the following identity:

\[
\int_{y=0}^{y=\ell} yf_\ell(y)dy + \int_{y=0}^{y=\ell} yf_\ell(y)dy = \ell
\]

where

\[
\tilde{f}_\ell(y) = f_\ell(\ell - y) \quad y \in (0,\ell).
\]

In words, the expected distance measured from \(i\), plus that measured from \(j\) equals the length of the link \(\ell\). This is a trivial but subtle identity. It is important when one moves across nodes and the associated parameters (C's and CL's) defining \(\bar{s}^2(x)\) change.
We will write
\[ y = \bar{y} \]
\[ \bar{y}_\xi(i \rightarrow j) = \int_i^j y f_\xi(y) dy \]
\[ y = 0 \]
and
\[ \bar{y}_\xi(j \rightarrow i) = \int_i^j y f_\xi(y) dy \]
\[ y = 0 \]

The following lemma describes the behavior of \( s^2(x) \) across nodes in a tree network:

**Lemma 5.3.2:** \( s^2(x) \) is convex along any path in a tree network.

**Proof:** Proof techniques are similar to earlier proofs on nodal-demand-only networks. Since \( s^2(x) \) is convex on a link, we need only concern ourselves with the behavior of \( s^2(x) \) across nodes.

Consider a specific path \( i - j - k \). We need to prove:

\[ s^2(j)_{\text{in}} \leq s^2(j)_{\text{out}} \]
along \( i-j-k \).

We denote the associated parameters and partitions with ('') and (""") when we are on link \( (i,j) \) and link \( (j,k) \) respectively. The length of \( (i,j) \) is \( \bar{\lambda}' \) and that of \( (j,k) \) is \( \bar{\lambda}"" \). \( \Delta \) and \( \Delta L \) are defined in Section 5.2.2. It is not difficult to verify that one can relate the relevant ('') and (""") parameters as follows:

\[ C_4'' = C_4' + 2 \sum_{p \in \Delta} h d(p,j) + \bar{\lambda}' \sum_{p \in N} h \]
\[ C_6'' = C_6' + 2 \sum_{p \in \Delta} h \bar{w} \]
\[ \begin{align*}
\text{CL}_4'' &= \text{CL}_4' + 2 \sum_{\ell \in \Delta L} f_\ell d(\lambda(a),j) + \bar{\lambda}' \sum_{\ell \neq \ell'} f_\ell \\
\text{CL}_6'' &= \text{CL}_6' + 2 \sum_{\ell \in \Delta L} f_\ell \bar{w}_\ell + 2f_\ell \bar{w}_\ell' \\
\text{CL}_8'' &= \text{CL}_8' + 2 \sum_{\ell \in \Delta L} f_\ell \bar{y}_\ell + [\bar{y}_\ell, (i + j) + \bar{y}_\ell, (j + i)] f_\ell \\
&= \text{CL}_8' + 2 \sum_{\ell \in \Delta L} f_\ell \bar{y}_\ell + f_\ell \bar{y}_\ell'
\end{align*} \]

where the node set \( \Delta \) and link set \( \Delta L \) are defined in Section 5.2.2.

\[ \begin{align*}
\bar{s}^2(j)'_{\text{out}} &= 2\beta^2/v^2 x \left|_{x=0} \right. + 2\beta^2/v^2 c_4'' + 2\beta/v \ c_6'' + 2\beta^2/v^2 \text{CL}_4'' + 2\beta/v \ \text{CL}_6'' + 2\beta^2/v^2 \text{CL}_8'' \\
&+ 4\beta/v \ f_\ell \bar{w}_\ell'' \bar{F}_\ell''(x) \bigg|_{x=0} \\
&= 2\beta^2/v^2 c_4' + 2\beta/v \ c_6' + 2\beta^2/v^2 \text{CL}_4' + 2\beta/v \ \text{CL}_6' + 2\beta^2/v^2 \text{CL}_8' \\
&+ 2\beta^2/v^2 \ell' \left[ \sum_{p \in \Delta} \sum_{\ell \in L} f_\ell + f_\ell', \right] + 4\beta/v \ f_\ell \bar{w}_\ell, \\
&+ [4\beta^2/v^2 \sum_{p \in \Delta} \sum_{\ell \in L} f_\ell d(p,j) + 4\beta/v \sum_{\ell \in \Delta L} \bar{w}_\ell + 4\beta^2/v^2 \sum_{\ell \in \Delta L} f_\ell d(\lambda(a),j) + 4\beta/v \sum_{\ell \in \Delta L} f_\ell \bar{w}_\ell + \\
&+ 4\beta^2/v^2 \sum_{\ell \in \Delta L} f_\ell \bar{y}_\ell] \\
&= 2\beta^2/v^2 x \bigg|_{x=\bar{x}} + \bar{s}^2 + 4\beta/v \ f_\ell \bar{w}_\ell, \bar{F}_\ell'(x) \bigg|_{x=\bar{x}} + \text{positive constant} \\
&= \bar{s}^2(j)'_{\text{in}} + \text{positive constant} \geq \bar{s}^2(j)'_{\text{in}}.
\end{align*} \]
Lemma 4.2.8 applies to $\bar{s}_2(x)$ because $\bar{s}_2(x)$ is convex along any path; namely, when $\bar{s}_2(x)$ starts increasing in a direction, it will never decrease again.

5.3.2 Properties of $\overline{TR}$ on a Tree Network:

The objective of this section is to prove that $\overline{TR}$, when finite, is convex along any path on a tree network with continuous link demands. Methods of proof are similar to those in Chapter 4. Continuous link demands, however, introduce non-polynomial (of x) terms in $\overline{t}$, $\overline{s}$ and $\overline{s}^2$, which make the analysis more complicated. Since the contribution from nodal and link demands to $\overline{TR}$ are non-additive, we cannot use the methodology of the minisum problem in Section 5.2. Since proving the convexity of $\overline{TR}$ across a node is easy, we have tried to consider infinitesimal movement, $\Delta x$, along a link and imagine that we are crossing a "node" with weight $f(x)\Delta x$ (when $f(x)$ is the demand probability density function). However, the functions $\overline{t}$, $\overline{s}$ and $\overline{s}^2$ change across $x$, whereas they remain constant across a node. Taking the limit of $\Delta x$ going to zero leaves us with the evaluation of $\overline{TR}(\cdot)$". Therefore, we are forced to take the direct approach of showing the positivity of $\overline{TR}(\cdot)$". The method of proof is that of "divide-and-conquer". We will divide the expression for $\overline{TR}(\cdot)$" into several components and prove that each of them is non-negative. The proof of convexity of $\overline{TR}$ across a node parallels that in the case of nodal-demand-only, given the convexity of $\overline{s}^2$ and $\overline{t}$. First, we will derive the algebraic expression for $\overline{TR}(\cdot)$". The analysis that follows is lengthy and we will provide a recap at the end of the discussion. We will again
suppress the argument \( x \) in \( \overline{s^2}, \overline{s}, \overline{t} \) and their derivatives.

\[
\frac{d^2 \overline{TR(x)}}{dx^2} = \frac{\lambda}{2} [1-\lambda s]^{-3} [N(x,\lambda)] + \frac{\lambda^2}{2} [1-\lambda s]^{-2} \beta \overline{s^2} \overline{t} + \overline{t}
\]

where

\[
N(x,\lambda) = N_1(x)\lambda^2 + N_2(x)\lambda + N_3(x)
\]

and

\[
N_1(x) = (s)^2 \overline{s^2''} - 2\beta \overline{s s'}^{2''} + 2\beta \overline{s^2} (t')^2
\]

\[
N_2(x) = 2\beta \overline{s^2} t' - 2s \overline{s^2''}
\]

\[
N_3(x) = \overline{s^2''}
\]

\[
t'' = \frac{2}{v} \int_{m}^{\infty} f(x) > 0,
\]

where the facility is at \( x \) on \( (a,b) = m \).

Since we are only concerned with the location at which \( 1 - \lambda s > 0 \), and \( \frac{\lambda^2}{2} (1-\lambda s)^{-2} \beta \overline{s^2 t''} + \overline{t''} > 0 \), \( \overline{TR(x)}'' > 0 \) if we can show that \( N(x,\lambda) > 0 \). The reason for separating out terms involving \( t'' \) is that \( t'' \) may be zero (if density is zero at \( x \)) and will not contribute to the positiveness of \( \overline{TR(x)}'' \).

Had there been no link demands, substituting \( \overline{s^2''} = 2\beta^2/v^2 \) gives us exactly the same expression for \( \overline{TR(x)}'' \) as the nodal-demand-only case. The next step now is to separate the non-polynomial terms (of \( x \)) in \( \overline{s^2}, \overline{s^2'} \) and \( \overline{s^2''} \) from \( N(x,\lambda) \) and prove positivity of the two components.
From Section 5.3.1, we can write $s^2(x)$ as

$$s^2(x) = \frac{\beta^2}{\nu^2} x^2 + s_0 x + s_0' + 4\beta/\nu \int_0^x \gamma_m w_m[F_m(x) - \int_0^x y f_m(y)dy]$$

$$(s^2(x))' = 2\beta^2/\nu^2 x + s_0' + 4\beta/\nu \int_0^x \gamma_m w_m F_m(x)$$

$$(s^2(x))'' = 2\beta^2/\nu^2 + 4\beta/\nu \int_0^x \gamma_m w_m F_m(x).$$

Let

$$\hat{s}^2(x) = \frac{\beta^2}{\nu^2} x^2 + s_0^2 x + s_0^2'$$

$$\hat{s}^2(x)' = 2\beta^2/\nu^2 x + s_0^2'$$

$$\hat{s}^2(x)'' = 2\beta^2/\nu^2.$$

$N(x, \lambda)$ can be written as

$$N(x, \lambda) = N_1(x)\lambda^2 + N_2(x)\lambda + N_3(x) + 4\beta/\nu \int_0^x \gamma_m w_m M(x, \lambda)$$

where

$$\hat{N}_1(x) = (s)^2 s^{2'} - 2\beta \hat{s} s^{2'} t + 2\beta^2 \hat{s} t^2$$

$$\hat{N}_2(x) = 2\beta \hat{s}^{2'} t - 2 \hat{s} s^{2''}$$

$$\hat{N}_3(x) = s^{2''}.$$

and

$$M(x, \lambda) = [M_1(x)\lambda^2 + M_2(x)\lambda + M_3(x)]$$

$$M_1(x) = (s)^2 f_m(x) - 2\beta \hat{s} t F_m(x) + 2\beta^2 \hat{t}^2 K_m(x).$$
\[ M_2(x) = 2\beta \bar{s} F_m(x) - 2s f_m(x) \]

\[ M_3(x) = f_m(x) \]

\[ K_m(x) = x F_m(x) - \int_0^x y f_m(y) dy. \]

Collecting terms, we have:

\[ \begin{align*}
M(x, \lambda) &= [f_m(x)(1-\lambda \bar{s}) - f_m(x)\lambda \bar{s}(1-\lambda \bar{s}) + 2\beta \bar{t}' F_m(x) \lambda (1-\lambda \bar{s}) + 2\beta^2 (\bar{t}')^2 K_m(x) \lambda^2] \\
&= [f_m(x)(1-\lambda \bar{s})^2 + 2\beta \bar{t}' F_m(x) \lambda (1-\lambda \bar{s}) + 2\beta^2 \lambda^2 (\bar{t}')^2 K_m(x)]
\end{align*} \]

because

\[ \int_0^x y f_m(y) dy < x \int_0^x f_m(y) dy = x F_m(x). \]

Also note that \( \bar{t}' > 0 \) when one moves down the tree from the minisum location. Therefore,

\[ M(x, \lambda) \geq 0 \text{ for } \lambda \in (0, 1/\bar{s}) \]

We still have to show the positiveness of

\[ \hat{N}_1(x) \lambda^2 + \hat{N}_2(x) + \hat{N}_3(x). \]

We compute

\[ \begin{align*}
\hat{N}_2(x)^2 - 4\hat{N}_1(x) \hat{N}_3(x) &= 4\beta^2 (\bar{t}')^2 [(s^2')^2 - 2 s^2 s^2] \\
&= 4\beta^2 (\bar{t}')^2 [(s^2')^2 - 4\beta^2 / v^2 \bar{s}^2].
\end{align*} \]
Recalling the expressions of $s'^2$ and $s^2$, we compute $(s'^2(x))^2 - 4s^2/s^2(x) = (s'^2_0)^2 - 4s^2/s^2_0$ to be independent of $x$. This is non-positive if we can show that

$$s^2(x) = \frac{\beta^2}{v^2} x^2 + s^2_0 x + s^2_0 > 0$$

for all $x$.

We will do this by following the derivation of $s^2(x)$,

$$s^2(x) = s^2(x) + 4\beta/v \int_0^x f_m \bar{w}_m [F_m(x) - \int_0^x y f_m(y) dy].$$

$s^2(x)$ consists of contributions from:

(i) nodal demands;

(ii) link demands (except link $m = (a,b)$);

(iii) demand on link $m = (a,b)$.

Nodal demands = \[ \sum_{j \in A} h_j \left[ w_j + \beta/v (d(a,j)+x) \right]^2 + \sum_{j \in B} h_j \left[ w_j + \beta/v (d(a,j)-x) \right]^2 \]

Link demands = \[ \sum_{\ell \in AL} \int_0^x \left[ w_{\ell} + \beta/v (d(\ell(a),a)+y+x) \right]^2 f_\ell(y) dy + \]
\[ + \sum_{\ell \in BL} \int_0^x \left[ w_{\ell} + \beta/v (d(\ell(a),a)+y-x) \right]^2 f_\ell(y) dy \]

Nodal and link contributions to $s^2(x)$ (except $m = (a,b)$) are polynomial in $x$ and are positive for all values of $x$. Demands from link $(a,b) = f_m \left[ \int_0^x \left[ w_m + \beta/v (x-y) \right]^2 f_m(y) dy + \int_x^m \left[ w_m + \beta/v (y-x) \right]^2 f_m(y) dy \right]$. It can be broken into polynomial terms in $x$ and "others". Polynomial terms will contribute to $s^2(x)$.
Polynomial terms of demands from link (a,b) = \( f_m [\beta^2/y^2 x^2 - (2\beta^2/y^2 y_m + 2\beta/v y_m) x + \beta^2/y^2 y_m + 2\beta/v w_m + w_m] \), is positive for all values of \( x \) if and only if

\[
(2\beta^2/y^2 y_m + 2\beta/v w_m)^2 - 4\beta^2/v^2 (w_m^2 + \beta^2/y^2 y_m^2 + 2\beta/v w_m y_m) \leq 0
\]

which is true because \( y_m^2 - (y_m)^2, w_m^2 - (w_m)^2 \) are the variances of respective random variables.

We have shown that the polynomial terms of \( s^2(x) \) include contributions from (i) nodal demands, (ii) link demands (except link (a,b) = m), and (iii) demands from (a,b) -- each of which is positive for all values of \( x \). Therefore,

\[
s^2(x) > 0 \quad \text{for all } x
\]

and

\[
(s_0')^2 - 4\beta^2/v^2 \frac{s}{s_0} < 0.
\]

Recap: Starting with the second derivative of \( \overline{TR}(x) \) with respect to \( x \):

\[
\frac{d^2 \overline{TR}(x)}{dx^2} = \frac{\lambda}{2} [1-\lambda s]^{-3} N(x,\lambda) + \frac{\lambda^2}{2} (1-\lambda s)^{-2} \beta \overline{s^2} t'' + t''
\]

Our attention is on locations of \( G \) with finite \( \overline{TR} \); i.e., \( (1-\lambda s) > 0 \).

Terms involving \( t'' \) are positive. To show positiveness of \( N(x,\lambda) \), we have separated out the non-polynomial (of \( x \)) terms in \( s^2, s' \) and \( s'' \) and proved them to be positive. The polynomial terms (both in \( \lambda \) and in \( x \)) are positive after we have proved that the "radical" of the quadratic (in \( \lambda \)) function is non-positive. That was done by recognizing that the
contribution from nodal and link demands to the polynomial terms in $s^2(x)$ is positive for all values of $x$. The above discussion leads to:

**Lemma 5.5.3:** $\overline{\text{TR}}(x)$, when finite, is convex on any link down a tree rooted at the minisum location.

This is without loss of generality because we can always place a fictitious node with zero weight at the minisum location when it is on the interior of a link. We can then perform the same rooting operation as was done in Chapter 4. Following a similar argument to prove convexity of $\overline{\text{TR}}(x)$ along a path on a tree with nodal demand only, we have:

**Lemma 5.3.4:** $\overline{\text{TR}}(x)$, when finite, is convex on any path along nodes of increasing depth in a tree rooted at the minisum location.

**Proof:** Since $\overline{\text{TR}}(x)$ is convex on a link down a tree rooted at the minisum location, we need only show that along $i-j-k$:

$$\overline{\text{TR}}(j)^\prime_{\text{out}} > \overline{\text{TR}}(j)^\prime_{\text{in}}$$

$$\overline{\text{TR}}(j)^\prime_{\text{out}} = \frac{\lambda}{2} [1 - \lambda \overline{s}(j)]^{-2} [s^2(j)^\prime_{\text{out}} (1 - \lambda \overline{s}(j)) + \lambda \beta \overline{t}(j)^\prime_{\text{out}} \overline{s^2}(j)] + \overline{t}(j)^\prime_{\text{out}}$$

since

$$1 - \lambda \overline{s}(j) > 0$$

$$\overline{s}(j), s^2(j)$$ continuous at node $j$

$$\overline{s^2}(j)^\prime_{\text{out}} > \overline{s^2}(j)^\prime_{\text{in}}$$

$$\overline{t}(j)^\prime_{\text{out}} > \overline{t}(j)^\prime_{\text{in}}$$

we have

$$\overline{\text{TR}}(j)^\prime_{\text{out}} > \overline{\text{TR}}(j)^\prime_{\text{in}}$$
In the next section, we will examine the appropriateness of the Trim and Search Algorithms as applied to the case of a tree network with continuous link demands. We will also trace the trajectory of the SQM when the total network intensity $\lambda$ is varied.

5.4 Locating the SQM and Parametric Analysis on a Tree Network with Continuous Link Demands

5.4.1 Locating the SQM:

With loss of generality, we can assume that the minisum location is located at a node. If it lies on the interior of a link, we can always place a fictitious node there and assign it a weight of zero. The weights and density functions of the two newly created links can be easily adjusted. Therefore, as before, we will root our tree network at the minisum location. Due to the convexity nature of $t(x)$ on a tree network, $t(x)' > 0$ when we move down this rooted tree.

We recall from Chapter 4 that all the developments require only the evaluations of $\overline{TR}(x)'$, $\overline{t}(x)'$, $\overline{s^2}(x)$, $\overline{s^2}(x)'$ and do not depend on the exact form of these functions. The heart of the analysis lies in the facts that $\overline{TR}(x)$, $\overline{t}(x)$ and $\overline{s^2}(x)$ are convex and $\overline{t}(x)' > 0$ down the rooted tree. All the conditions are met exactly in the case of a tree network with continuous link demands. The only departure is when one has to locate the SQM on the interior of a link. In that instance, we solve the following equation (for $x$):

$$A(x, \lambda) = 0$$

In the case of nodal-demand-only, $A(x, \lambda)$, for fixed $\lambda$, is quadratic in $x$, thus we can solve for $x$ easily.
With continuous link demands, we no longer have a quadratic function. The functional form of $A(x,\lambda)$ depends on the demand probability density functions. We will have to use a numerical routine to find the roots of $A(x,\lambda) = 0$. The constant $C_1$ in all the equations should be replaced by $\bar{t}(x)'$. The $\lambda$ independent test $(s^2')^2 - 8\beta/v^2(t')^2s^2 > 0$ can be readily applied to the Trim Algorithm also. We can apply the same analysis and algorithms exactly to a tree network with continuous link demands. This is a very pleasing situation, to be able to extend the same methodology to a much more general problem.

5.4.2 Parametric Analysis of the SQM as $\lambda$ Varies

We perform all the parametric analysis using $A(x,\lambda)$ and the $DTX(x,\lambda)$ profiles. Instead of looking at them as a function of $x$, which are quadratic for $A(x,\lambda)$ in the nodal-demand-only case, we treat them as a function of $\lambda$ for fixed values of $x$. $A(x,\lambda)$ is still quadratic in $\lambda$; $DTX(x,\lambda)$ maintains the same functional form when we replace $C_1$ by $\bar{t}(x)'$, which is positive down the rooted tree. The fact that the $DTX(x,\lambda)$ profiles in $\lambda$ space do not intersect hinges on the convexity of $TR(x)$ only, and not on its functional form. Figures 4.3.3 and 4.3.4 apply exactly to our present situation. The only difference is that in the nodal-demand-only situation, $\bar{t}' = C_1$ for all points on the same link. Thus the $DTX(x,\lambda)$ profiles intersect at $\lambda = 0$. However, in the presence of continuous link demands, $\bar{t}'$ is no longer a constant for $x$ on the same link. Convexity of $\bar{t}$ implies that $\bar{t}'$ will increase down the tree. Therefore, the $DTX(x,\lambda)$ profiles may not even intersect at $\lambda = 0$. 
5.5 A Numerical Example: The SQM on a Tree Network with Continuous Link Demands

We will construct a numerical example to illustrate the essence of our trim and search procedures. We also compute the range of $\lambda$ over which each node (and each link) is the optimal median. Figure 5.5.1 shows the tree network (together with all the relevant parameters) under examination. The tree is rooted at the minisum location, which happens to be a node (node 2). However, there is no loss of generality since we can always place a fictitious node of zero weight at the minisum location. We assume a uniform demand density function for each link over which the arrival rate is non-zero. Table 5.5.1 shows the value of $s^2$ at some of the nodes.

We now apply one version of our trim algorithm, testing the value of $s^2$ at each node in order of increasing depth from node 2.

Details of the trim procedure are as follows:

1. At node 2, $s^2 = 1402$.
2. At node 3, $s^2 = 1616$, immediate successor of node 2. 
   At node 1, $s^2 = 1223$, 
   We eliminate link (2,3) and node 3.
   We move down from node 1 to node 4.
3. At node 4, $s^2 = 879$.
   We move down from node 4 to its immediate successors.
4. At node 10, $s^2$ is very large.
   Eliminate branch 10.
   At node 5, $s^2 = 936$, but $s^2(4)'_{out} = -33.28 < 0$ along link (4,5)
   Therefore we have to include link (4,5), but we can eliminate branch 5.
Figure 5.5.1 Numerical Example: SQM on a Tree Network with Continuous Link Demands.
<table>
<thead>
<tr>
<th>Node j</th>
<th>2</th>
<th>3</th>
<th>1</th>
<th>4</th>
<th>10</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s^2(j)$</td>
<td>1402</td>
<td>1616</td>
<td>1223</td>
<td>879</td>
<td>large</td>
<td>936</td>
</tr>
</tbody>
</table>
The Median Seeking Path (MSP) is the residual subtree 2-1-4-5. Figure 5.5.2 shows this MSP.

To perform the search procedure, we need to know the specific value of $\lambda$. Instead of locating the SQM for only one value of $\lambda$, we perform parametric analysis to find all the ranges of $\lambda$. Table 5.5.2 lists all the relevant data for computing the ranges of $\lambda$. The equation we use is:

$$A(x, \lambda) = \frac{\beta}{\nu} t s + \frac{1}{\nu} t (s) - \frac{1}{\nu} t s + \frac{4}{\nu} t s |s|^2 + \frac{2}{\nu} t$$

where we have suppressed the argument $x$ in $s$, $s'$, $s''$, and $t'$.

Using Table 5.5.2, and solving $A(x, \lambda)$ for $\lambda$, we have the following range parametric for $\lambda$ (refer to notations in Section 4.4).

$$A(2^-) = \{ \lambda_1 \}; \quad A(2^+) = \{ \lambda_{10} \}$$

$$A(1^-) = (\lambda_2, \lambda_3); \quad A(1^+) = (\lambda_8, \lambda_9)$$

$$A(4^-) = (\lambda_4, \lambda_5); \quad A(4^+) = (\lambda_6, \lambda_7)$$

$$A(2,1) = (\lambda_1, \lambda_2); \quad A(1,4) = (\lambda_3, \lambda_4)$$

$$A(4, x_m) \cup A(x_m, 4) = (\lambda_5, \lambda_6)$$

$$A(4, 1) = (\lambda_7, \lambda_8); \quad A(1, 2) = (\lambda_9, \lambda_{10})$$

where

$$\begin{align*}
\lambda_1 &= 0 & \lambda_2 &= 0.0004857 \\
\lambda_3 &= 0.001524 & \lambda_4 &= 0.003388 \\
\lambda_5 &= 0.011406 & \lambda_6 &= 0.02457 \\
\lambda_7 &= 0.03373 & \lambda_8 &= 0.0366678 \\
\lambda_9 &= 0.03828 & \lambda_{10} &= 0.03925
\end{align*}$$
Figure 5.5.2  MSP Associated with Figure 5.5.1.
### TABLE 5.5.2

Values of $s'^{-2}$, $s$, $t$

<table>
<thead>
<tr>
<th>Node $j$</th>
<th>2</th>
<th>1</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s'^{-2}$</td>
<td>out</td>
<td>in</td>
<td>out</td>
</tr>
<tr>
<td></td>
<td>-97.44</td>
<td>-81.36</td>
<td>-81.20</td>
</tr>
<tr>
<td>$t'$</td>
<td>0</td>
<td>-0.02</td>
<td>-0.06</td>
</tr>
<tr>
<td>$s$</td>
<td>25.48</td>
<td>25.52</td>
<td></td>
</tr>
</tbody>
</table>
Each shaded area corresponds to DTX profiles of points on the same link.

DTX Profiles of Numerical Example in Figure 5.5.1.

Figure 5.5.3

DTX Profiles of Numerical Example in Figure 5.5.1.
Note that $x_m$ is the point on link (4,5) where $(s')^2 - 8\beta/v^2(\bar{r})^2s^2 = 0$. We solve this equation for $x_m$ and find $x_m = 0.96$. We sketch the DTX profiles in Figure 5.5.3 to show the relationship among the $\lambda$'s.

We can find the optimal location given any value of $\lambda \in (\lambda_1, \lambda_{10})$ by referring to Figure 5.5.3 and solving $A(x,\lambda) = 0$ for $x$, knowing on which link $x$ lies.
6.1 Preliminary

We have indicated in Chapter 2 some issues and important system elements one should consider in queueing-location problems. The potential for analytically interesting and practically useful research problems is very promising. Relative lack of work in this area prompts us to identify relevant models with potential real world applications. By taking various combinations of the system descriptors discussed in Chapter 2, one can formulate meaningful models with increasing complexity. In this chapter, as a first step in initiating future research in this area, we will formulate several potentially interesting and tractable problems selected from the broad spectrum of queueing-locational models. After each formulation, we try to make some intelligent guesses about possible solution techniques. Each model formulated here is defined on an undirected general network with discrete nodal demands.

6.2 Problem Formulations

Several interesting models will be formulated and solution procedures will be proposed in this section. We feel that this is one of the most exciting phases of a research process when one can speculate and "fantasize" about problem structure and analytical behavior of a model. The proposed solution methodology after each formulation will be informal and sometimes wild and unstructured -- welcome to the speculative stage of an exciting research paradise.
6.2.1 A Priority Queue Location Problem:

We formulate a stochastic queue median system with priority-oriented queueing discipline in this section. Under this queueing discipline, one categorizes customers into K priority classes, indexed by k; the lower value of k, the higher the priority. The selection rule for waiting customers to receive service is FCFS by priority class. We assume no interruption of service upon arrival of high priority customers. Again, we only concern ourselves with independent, time-homogeneous Poisson demands. The objective is to locate a single facility minimizing the sum of weighted (by priority class) average response time.

In addition to the usual network structure and arrival rate $\lambda$, we have to introduce variables and parameters which are priority specific. They are defined as follows:

- $p_{ik} = \text{fraction of total calls from node } i \text{ of priority } k; \text{ for } i \in N \text{ and } k = 1,2,...,K$
- $f_k = \sum_{i \in N} p_{ik} = \text{fraction of priority } k \text{ calls; } k = 1,2,...,K$
- $h_i = \sum_{k=1}^{K} p_{ik} = \text{fraction of calls from node } i, \text{ } i \in N$
- $\bar{w}_{ik} = \text{expected non-travel related service time of priority } k \text{ calls from node } i; \text{ } i \in N, k = 1,2,...,K$
- $v_k = \text{travel speed answering priority } k \text{ calls; } k = 1,2,...,K$
For a facility located at $x$ on $G$, we define the following:

- $t_k(x) = \frac{1}{f_k} \sum_{i \in N} p_{ik} d(x,i)/v_k$

  = expected travel time to answer priority $k$ calls

- $s_k(x) = \frac{1}{f_k} \sum_{i \in N} p_{ik} (\overline{w_{ik}} + \beta d(i,x)/v_k)$

  = expected service time of a priority $k$ call

  = $\overline{w_k} + \beta t_k(x)$

where

- $\overline{w_k} = \frac{1}{f_k} \sum_{i \in N} p_{ik} \overline{w_{ik}}$

- $s_k^2(x) = \frac{1}{f_k} \sum_{i \in N} p_{ik} (\overline{w_{ik}} + \beta d(i,x)/v_k)^2$

  = second movement of a priority $k$ call service time

- $s^2(x) = \sum_{k=1}^{K} f_k s_k^2(x)$

The above model operates as an $M/G/1$ system with $K$ priority classes. It is well known [ ] that the queueing time for the $k^{th}$ priority class is:

$$Q_k(x) = \frac{\lambda}{2} s^2(x) (1-\sigma_{k-1})^{-1} (1-\sigma_k)^{-1} \text{ if } 1 - \sigma_k > 0$$

$$= \infty \text{ otherwise}$$

where

$$\sigma_k = \sum_{j=1}^{k} f_j \lambda \overline{s_k(x)}$$

Average response time for a priority $k$ customer is

$$\overline{TR}_k(x) = Q_k(x) + t_k(x)$$
Our objective is to find $x$ on $G$, such that $\overline{TR}(x)$ is minimized, where

$$\overline{TR}(x) = \sum_{k=1}^{K} \gamma_k \overline{TR}_k(x)$$

where $\gamma_k$ are given positive constants reflecting the importance we attach to priority $k$ customers.

We will now make some speculative remarks about possible solution techniques to the above priority queue location problem. We note that $s^2$ is the same in each $Q_k$. $s^2$ takes on essentially the same functional form as that in the no-priority (or one priority) queueing median case, except for the difference in travel speed $v_k$ of priority $k$. $s^2$ will again be convex (and quadratic) in $x$ within a primary region. The only complicating factor is in the denominator of $Q_k$, which consists of two terms: $1 - \sigma_k$ and $1 - \sigma_{k-1}$. $\sigma_k$ can be interpreted as the aggregated system utilization factor ($= \lambda \sum_{i=1}^{K} \sum_{i=1}^{K} s_i(x)$) generated from priority one through $k$. A natural starting point (in minimizing $Q_k$) is from the minisum-k location -- a point on the network which minimizes $\sigma_k$. Another starting point is the minisum (k-1) location. An interesting question to ask is: Will the minimizer of $Q_k$ lie on a path between these two minisum locations? These two minisum points are nodal because the functions $\sigma_{k-1}$ and $\sigma_k$ are weighted (positively) sums (by priority) of concave functions on a link. Therefore, for each $Q_k$, we have these functions that one wishes to minimize: $\sigma_k$, $\sigma_{k-1}$ and $s^2$. $s^2$ is independent of the index $k$.

A second interesting question is: How do the weights $\gamma_k$ 's affect the optimal priority queue median location? How would one locate all the efficient points as the $\gamma_k$'s vary? Efficient points are defined in the traditional sense of Pareto-optimality: a point $x$ on $G$ is Pareto-optimal
if and only if there does not exist a point \( y \) on \( G \) such that \( \bar{T}_{R_k}(y) \leq \bar{T}_{R_k}(x) \) for all \( k \), with at least one of the inequalities holding strictly.

A natural restriction to impose is: \( Y_i \geq Y_j \), if \( i \leq j \), since one would like to attach more weights to higher priority customers. When \( Y_k = 0 \), one can interpret this as "pruning" away class \( k \) customers. When all \( Y_k \)'s are strictly positive, we would be interested in point \( x \) such that \( 1 - \lambda s > 0 \), where \( s \) is the system-wide travel time averaged over all priority classes. A natural starting point to search for the optimal location is at the Hakimi median of \( G \).

6.2.2 **A Selective Pruning Problem**

This problem incident is a mixture of stochastic queue and loss systems. A selected subset of nodes are designated as loss nodes. Service requests from loss nodes will be answered by back-up units at a cost \( Q \) regardless of server status. The remaining system will operate as a stochastic queue system. One would like to find the optimal partition so that the sum of the weighted (by the proportion of loss and queued demands) response time and the cost of loss is minimized. We have the following additional notations:

For a subset \( J \subseteq N \), we define:

\[
H(J) = \sum_{i \in J} h_i
\]

and

\[
\bar{J} = N - J
\]
For a facility located at $x$ on $G$:

$$
\overline{t}(x\mid J) = \frac{1}{H(J)} \sum_{j \in J} h_j d(x, j) / v
$$

$$
\overline{w}(J) = \frac{1}{H(J)} \sum_{j \in J} \overline{w}_j
$$

$$
\overline{s}(x\mid J) = W(J) + \beta \overline{t}(x\mid J)
$$

$$
\overline{s}^2(x\mid J) = \frac{1}{H(J)} \sum_{j \in J} [\overline{w}_j + \beta d(x, j)]^2
$$

$$
\lambda(J) = H(J) \lambda
$$

The response time for the stochastic queue system operating on a subset of nodal demands $h_j$, $j \in J$ is:

$$
\overline{TR}(x\mid J) = \frac{\lambda(J)}{2} \overline{s}^2(x\mid J) [1 - \lambda(J) \overline{s}(x\mid J)]^{-1}
$$

$$
= \infty \quad \text{otherwise}
$$

Our objective is to find a partition of the node set $(J$ and $\overline{J})$, such that the following objective is minimized:

$$
\min_{J \subseteq N} \min_{x \in G} \{H(J) \overline{TR}(x\mid J) + [1 - H(J)]Q\}
$$

This problem is potentially very difficult to solve because the set $J$ may not even be comprised of contiguous nodes, and there are about $2^n$ possible partitionings of the set $N$, $n = |N|$. 
The major difficulty that lies with this problem is that the remaining nodes (after pruning) may not even be contiguous. This can be seen intuitively as follows: there may exist a node with extremely large $h_j$; as $\lambda$ increases, the inclusion of this node in the queueing component may leave the system infeasible (i.e., infinite waiting time). The pruning of this "busy" node can leave the remaining nodes "non-contiguous".

There are several heuristics one may consider, one of which is to pruning away outliers. Outliers can be defined loosely as nodes which are "far away" from the Hakimi median. One may use exchange heuristics to "swap" pruned and unpruned nodes. Another difficulty here is that the parameters on a primary region (i.e., $c$'s) as defined in Chapter 4 will change after each exchange. A more systematic analysis of the changes in the parameter $c$'s is necessary to make any exchange heuristic operational -- in terms of marginally improving the objective function.

We will introduce a more simple-minded pruning problem, which shows more promise of tractability, in the next section.

6.2.3 A Uniform Pruning Problem:

In this formulation, a constant fraction of calls on all nodes are rejected at a cost $Q$. The remainder of the calls operate in a stochastic queue system environment. One can visualize this system as a Poisson-Bernoulli process: upon arrival of a call for service, a Bernoulli trial is performed to decide the manner under which this call is to be served, whether by a back-up unit at a cost $Q$, or by the primary unit in a stochastic queue environment. We have discussed this problem formulation in Chapter 4. We define
and we seek a Bernoulli probability \( p \) such that the following objective is minimized:

\[
(1-p)v((1-p)\lambda ) + pQ.
\]

As discussed in Chapter 4, this is a convex problem if \( v(*) \) is convex. The first research step in the solution of this problem is to prove the convexity of \( v(*) \). Suppose \( v \) is convex. Our search (for the optimal \( p \)) is reduced to finding the zero of:

\[
F(q) = v(q\lambda ) + q \lambda v'(q\lambda ) + Q
\]

where \( q = 1 - p \)

if \( F(1) < 0 \), then \( q = 1 \) is optimal;

if \( F(0) > 0 \), then \( q = 0 \) is optimal;

otherwise, one can find the zero of \( F(q) \) for \( 0 < q < 1 \).

The values of \( v(q\lambda ) \) and \( v'(q\lambda ) \) depend on the optimal location \( x \) on \( G \), where \( x \) and \( q\lambda \) satisfy \( A(x, q\lambda ) = 0 \) (see Section 4.3.1).

6.2.4 A Districting Problem with Infinite Queueing Capacity:

We understand that there are no known closed form results regarding the average behavior of an M/G/k system, where \( k \) is the number of servers. One can only hope to use bounds and approximations to analyze problems with light and heavy traffic intensity (\( \lambda \)). Therefore, it is natural to partition the network into \( k \) districts, each of which operates as an
independent $M/G/1$ system. The partition itself is bound to be difficult. We can only hope to utilize heuristic procedures and settle with "local" optimal solutions. With the same notation as in Section 6.2.2, our districting problem is:

$$\min_{J_i, i=1,2,\ldots,k} \min_{x_i \in G} H(J_i) \ TR(x_i | J_i)$$

where

$$\bigcup_{i=1}^{k} J_i = N$$

and

$$J_i \cap J_j = \emptyset \quad i \neq j; \ i=1,\ldots,k$$
$$j=1,\ldots,k.$$  

This districting problem is at least as difficult, in terms of the partitioning effort, as the selective pruning problem of Section 6.2.2. As an added complexity, one has to solve for $k$ independent stochastic median problems. As mentioned in Section 4.5.2, minimization of $s^2$ is the same as equalization of workload (at least in the case of a tree network). A natural partitioning is to design districts with "equal" utilization factors, i.e., equalizing $H(J_i) \ s(x_i | J_i)$, where $x_i$ is the optimal stochastic median of component $J_i$. However, one may use the respective Hakimi median of component $J_i$ as an approximation. Equalization of utilization factors takes into account, in addition to nodal weights $k$'s, the effect of travel time on the objective function. With this in mind, we formulate a districting problem in a stochastic loss environment (instead
of queue), believing the notion of equalizing the utilization factor may lead to optimal partitioning.

6.2.5 **A Districting Problem in a Stochastic Loss Environment:**

Our problem here is to partition the network into \( k \) districts, each of which operates optimally as an independent stochastic loss system (as in Chapter 3). The objective for each sub-system is to find a point \( x \) on \( G \), such that

\[
R(x|J_i) \quad \text{is minimized}
\]

where

\[
R(x|J_i) = [1 - P(x|J_i)] \overline{t}(x|J_i) + P(x|J_i)Q
\]

and

\[
P(x|J_i) =
\]

= Probability that the server of sub-system \( i \) is busy given that the facility is located at \( x \) on \( G \).

\[
\frac{\lambda H(J_i) \overline{s}(x|J_i)}{1 + \lambda H(J_i) \overline{s}(x|J_i)}
\]

where \( H(J_i) \) and \( \overline{s}(x|J_i) \) are defined in the same manner as in Section 6.6.2.

The overall objective is to partition the node set \( N \) into \( k \) non-overlapping subset \( J_i \), such that the aggregated average cost to the system is minimized. Mathematically, we have:
\[
\min_{J_i} \min_{x_i \in G} H(J_i) R(x_i | J_i)
\]
\[
i = 1, 2, \ldots, k
\]

where

\[
\bigcup_{i=1}^{k} J_i = N
\]

and

\[
J_i \cap J_j = \emptyset \quad \text{if } i \neq j; \quad i = 1, 2, \ldots, k,
\]
\[
j = 1, 2, \ldots, k.
\]

The results of Chapter 3 (equivalence of the minisum location and the stochastic loss median) prompt us to believe that given a set of \( k \) locations on the network, the optimal partition is to allocate each demand point (node) to its nearest service facility. This is a scheme in which we locate (facilities) first and then allocate (demand points). Also, we believe that a good (if not optimal) partition is to equalize the utilization factor \( (\lambda H(J_i) s(x_i | J_i)) \) of each district, where \( x_i \) is the optimal loss median of district \( J_i \). A solution procedure which combines these two observations should give us a good location-allocation scheme.

The last Chapter, Chapter 7, provides a summary of results (by Chapter) and a few concluding remarks.
Chapter 7

CONCLUSIONS AND SUMMARY OF RESULTS

7.1 Summary of Results

We will first summarize briefly our research effort of each chapter, before making a few concluding remarks to end this thesis. In Chapter 1, we discussed the need for research in the integration of queueing-like congestion into locational decisions. Chapter 1 also contains a literature review and a road map for the entire thesis.

Originally, we intended to provide a classification scheme for problem identification in Chapter 2. The intended scheme would categorize each problem formulation by a vector representation to capture all the essential system elements central to a queueing-location problem (as is done in the queueing literature, e.g., M/G/k system). However, we feel that more careful thinking and pioneering insights are needed before one can do justice to such a classification scheme. A hastily put together classification may suffer from lack of elegance, modularity and permanency. We will venture into such an ambitious effort after we have gained more insight into problem structure in the future. Therefore, we gathered and organized what we believe to be the essential features, central to a queueing-location problem, into six components. We hope that these components capture all the important system elements in most (if not all) of the potentially interesting research problems in this area. In other words, we stopped short of defining a classification device, but we have provided a solid starting base for such an effort in the future.

In Chapter 3, we generalize the stochastic loss one-median problem into an n-server-single-facility loss system. In this model, one wishes
to locate a single facility to house \( n \) servers such that the weighted sum of travel time (when some servers are free) and cost of rejection (when all servers are busy) is minimized. We were able to prove the equivalence of the minisum location and the \( n \)-server-single-facility loss location. This equivalence is true under any topological structure and demand distribution pattern.

Chapter 4 contains a complete analysis of the stochastic queue one median problem on a tree network with discrete nodal demands. In this model, arriving customers enter a queue with infinite capacity when the server is busy. The objective is to locate a facility to minimize the average response time (sum of queueing delay and travel time). We have proved certain convex properties of the average response time over a tree network. This convexity (of response time) plays an important role in the development of an efficient procedure to search for the optimal facility location. We also performed parametric analysis to study the trajectory of the optimal facility location when total system demand rate varies. A numerical example was constructed to demonstrate that multiple local minima (of the response time) can exist on the same link of a general network.

Chapter 5 generalized the results of Chapter 4 to a situation where one has, in addition to discrete nodal demands, continuous link demands governed by general probability density functions. All the convexity and parametric results generalized nicely to this problem. We also formulated and analyzed the minisum location problem on a general (and a tree) network with continuous link demands. It was done to provide a foundation for the analysis of the stochastic queue one median problem. We concluded
that, in a general network, an exhaustive search over all links and nodes may be needed to locate the minimum point. We have also provided a numerical example to demonstrate the procedure in the search for the stochastic queue median. Parametric analysis was also performed on the same example.

In Chapter 6 we formulated five queueing-location problems, each of which, we believe, is interesting and potentially analytically tractable. We have also speculated on possible solution procedures and problem structures for each of the formulated models in Chapter 6.

We will end this thesis with a few concluding remarks in the next section.

7.2 Concluding Remarks

The main objective of this thesis has been to merge the concerns of location theory and queueing theory. Little has been done towards the integration of these two areas. Relative lack of work in this research direction leaves us enormous freedom in problem selection and numerous opportunities for innovative solution methodology. The ultimate purpose is to introduce as much realism into a model with potential real world application, without leaving the problem intractable. We have incorporated a moderate degree of complexity in several instances of queueing-location problems in this thesis.

The issue of system elements discussed in Chapter 2 provides a guide for problem selections and future research activities. We understand that we have just begin to scratch the surface of a very rich research area and it is our intention to place due emphasis on location decisions in a stochastic environment. Future research efforts should provide us
with much insight into the interface between location and queueing theory, and the effect of stochastic congestion on locational decisions. We hope that we have been successful in sparking interest in this area by showing feasibility in problem formulation and solution methodologies. It is our belief that such research activity will eventually find its place in real world implementation.
REFERENCES


