MOUNTAIN-PLAINS CONSORTIUM
Transportation Research, Public Service & Education

MPC REPORT NO. 92-14

Statistical Methods for Optimally Locating Automatic Traffic Recorders

Chih-hsu Cheng
Christopher J. Nachtsheim
P. George Benson

July 1992

North Dakota State University
Fargo, North Dakota

Colorado State University
Fort Collins, Colorado

University of Wyoming
Laramie, Wyoming

Utah State University
Logan, Utah

Statistical Methods for Optimally Locating Automatic Traffic Recorders

Chih-hsu Cheng
Department of Management Sciences
The Ohio State University

Christopher J. Nachtsheim
Department of Operations and Management Science
University of Minnesota

P. George Benson
Department of Operations and Management Science
University of Minnesota

July 1992

Acknowledgement

This project was sponsored by the University of Minnesota, Center for Transportation Studies, U.S. Department of Transportation, and the Mountain-Plains Consortium. The financial support is acknowledged, but the authors assume complete responsibility for the contents herein.

Disclaimer

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

Table of Contents

List of Tables

List of Figures

EXECUTIVE SUMMARY

In this report, we present new, computer-based statistical methods for the optimal placement of Automatic Traffic Recorders (ATR). The goal of each method is to locate a set of ATRs so as to improve the overall efficiency and accuracy of Annual Average Daily Traffic (AADT) estimates. The precise estimation of AADTs is essential because of the important role they play in many highway design, maintenance and safety decisions.

Because of the huge number of potential ATR sites in a typical state highway system (e.g. 220,000), optimal selection of ATR sites is a very large combinatorial problem. Accordingly, site selection is currently accomplished through judgmental and/or design-based sampling techniques (e.g. random sampling). By developing fast and efficient computer algorithms to accomplish the purposive selection of an optimal sample, we demonstrate that model-based sampling is a viable alternative to classical design-based sampling techniques.

The algorithms developed in this project include an exchange algorithm and a two-stage sampling algorithm. In the rank-1 exchange algorithm, ATR sites are sequentially added to and deleted from the design. It generates highly efficient designs (i.e., sets of ATR sites) without exhaustively searching through all possible designs. In the two-stage sampling approach, similar sites are statistically clustered, then approximate design techniques are used to calculate the optimal weight for each cluster. Based on these optimal weights, a random sample of sites is selected from within each cluster. The speed of this two-stage sampling algorithm makes it an ideal approach for large-scale problems. Using traffic data provided by the Minnesota Department of Transportation, we demonstrate empirically that both algorithms are substantially better in terms of average variance of prediction than simple random sampling.

The software that implements the algorithms described in this report is available from the authors.

CHAPTER 1
INTRODUCTION

1.1 Introduction to Traffic Data Collection

Traffic monitoring involves the collection of traffic data such as traffic volume, traffic composition, vehicle speed, truck weight. These data provide information for highway planning, engineering design, and traffic control. In addition, legislative decisions such as budget allocation, selection of state highway routes, and the determination of speed limits require traffic data.

The traffic monitoring program recommended by the Federal Highway Administration consists of three elements or subprograms: Automatic Traffic Recording (ATR), the Highway Performance Monitoring System (HPMS), and a "special needs" programs. These three programs generate information about traffic volumes, traffic mix (i.e. vehicle classification), and truck weights. This project focuses on the problem of using the traffic-volume data provided by these programs to estimate traffic volumes on highways that are not currently monitored.

Traffic-volume estimation can be subdivided into two basic categories: Annual Average Daily Traffic (AADT) estimation, and Annual Vehicle Miles of Travel (AVMT) estimation. AADT is a point-specific measure while AVMT is a system measure. AADT is defined as the average number of vehicles that pass through a particular section of a road each day. AVMT of a state highway system is derived from multiplying each road section's AADT and section length and summing the product for all sections. Thus, the quality of AVMT estimation depends directly on the quality of AADT estimations.

In addition to estimating AVMT, AADT is used in many highway design, maintenance and safety decisions. These include: (1) design thresholds such as shoulder widths, lighting, and guardrail requirements; (2) estimation of accident rates per 1000 vehicles (used to rank highway improvement projects); and (3) estimation of peak hour volume, a statistic that is used in developing other estimators. Accordingly, the success or failure of highway planning efforts is to a certain degree contingent on the quality of AADT estimation techniques.

The most reliable approach to estimating AADT would be to place traffic-data collection equipment--such as Automatic Traffic Recorders (ATRs)--on every section of every road in the highway system in question. Currently, this approach is impractical , however, because of the expense involved in purchasing and installing ATRs. No state can afford an ATR on each road section. The state of Minnesota currently has 216,000 road sections and only 151 ATRs in place. For the sections that do not have ATRs, AADTs are estimated through other methods such as short-duration tube counts. This report focuses on the following two AADT-estimation issues:

(1) How can locations be identified for available ATR equipment that lead to better AADT estimates than are currently produced?

(2) If additional ATRs become available, where should they be placed?

Because ATRs are built into the pavement and cannot be easily relocated, the careful location of ATR sites is important. Random selection of ATR sites is always an option, but it ignores similarities and differences among roads and, therefore, could be quite inefficient. For example, two road sections that service a particular recreational area may have similar use-profiles. In such cases, it would be more efficient to place an ATR on just one of the two sections. Random sampling could result in both sections being selected.

In the state of Minnesota, the placement of ATR sites has been largely subjective in nature. Very little science has been brought to bear on the problem of site selection. This appears to be the case in most other states as well. In fact, the *Traffic Monitoring Guide* concedes that "in terms of statistical rigor, most continuous (ATR) programs lack a firm statistical base, their design can best be characterized as evolutionary and incremental."[1]

The purpose of this project was to investigate new, computer-based statistical methods for optimal placement of ATR sites. The primary technique that we investigated is called model-based sampling; it is reviewed in Chapter 3.

1.2 Challenges of This Research

This research is unique in that it deals with several challenges that have not been fully explored in the literature on model-based sampling. These challenges include:

(1) The size of the problem makes finding the optimal solution a non-trivial problem. In Minnesota, for example, there are more than 3.49 x 10^{540} possible ways to select 151 ATRs sites among 216,000 potential sites. The exhaustive search for the optimal solution is impossible even with the most powerful computer available today.

(2) Most existing model-based sampling research has focused on the problem of estimating population totals (or population averages) and the design criterion is usually to minimize the variance in predicting this population total. In our case, AADTs are point-specific measures. We are more interested in predicting individual AADTs, not "average AADTs" or "total AADTs". As a result, minimizing the maximum (minimax) or average variance in predicting these AADTs

[1] Traffic Monitoring Guide, Page 3-2-1.

appears to be a more reasonable criterion. The minimax criterion has not been fully-investigated in the literature and is difficult to deal with mathematically (and computationally.)

(3) Almost all model-based sampling techniques developed so far assume the data are independently and identically distributed. This assumption does not generally hold in this research. Traffic volumes on neighboring road sections are correlated, in ways that may be difficult to model.

This project attempted to find solutions to the above problems. The project report proceeds as follows:

(1) In Chapter 2, we provide background information about the traffic monitoring program currently used at the federal and state levels in the U.S. We also discuss problems with current AADT-estimation procedures and propose an alternative approach: model-based sampling.

(2) In Chapter 3, we review existing finite population sampling methodologies.

(3) In Chapter 4, we demonstrate the applicability of model-based sampling techniques to the AADT-estimation problem by developing an intuitive superpopulation model. We also explain the computational complexity of dealing with spatial correlation among data points in this problem, along with some possible solutions. The last part of this chapter is devoted to the identification of the variance function which plays an important role in the computational algorithms we present in Chapters 5 and 6.

(4) We discuss problems associated with the application of the minimax criterion and present an alternative optimality criterion in Chapter 5. Based on this criterion, we develop an exchange algorithm for finding the optimal location of ATRs. This exchange algorithm is highly effective when the problem is small to medium in size.

A variation of the exchange algorithm for populations of special structure is also discussed in this chapter.

(5) In Chapter 6, we present a more general two-stage sampling approach designed to deal with large problems in which the population has no special structure but can be meaningfully clustered. This newly developed algorithm generates nearly optimal samples with relatively little computational effort.

(6) In Chapter 7, we evaluate and provide recommendations regarding the algorithms developed in previous chapters in terms of the sampling efficiency and computer time needed to generate the optimal solution.

(7) Finally, in Chapter 8 we discuss the possible extensions of this research.

CHAPTER 2
ELEMENTS OF FEDERAL TRAFFIC MONITORING PROGRAM AND CURRENT PROCEDURES USED IN MINNESOTA

In this chapter, we give a more detailed description of the three elements of the federal traffic monitoring program: the Automatic Traffic Recorder (ATR) element, the Highway Performance Monitoring System (HPMS), and the special needs element. We also describe how the Minnesota Department of Transportation (Mn/DOT) implements these three elements. We then explain why the current procedure used by Mn/DOT is inadequate. Finally, we briefly explain why we focus on the ATR element in this research.

2.1 Automatic Traffic Recorder (ATR) Element

The ATR element, also referred to as the "continuous element" in the *Traffic Monitoring Guide*, is the backbone of contemporary state traffic counting programs. Continuous traffic counts are taken 365 days a year on a small number of sites by Automatic Traffic Recorders. (See Benson, Pisharody, and Yeldan [1985] for a survey of available traffic-data-recording devices.) These counts provide useful and reliable information for highway and traffic planning. One of the important uses of ATR data is to establish "seasonal traffic-volume factors" for different groups of roads. (See Benson, Pisharody, and Yeldan [1986] for a detailed description of the data collection and processing methods currently employed by Mn/DOT's traffic data unit.) These seasonal factors are used in the HPMS element (described in Section 2.2) to produce estimates of Average Annual Daily Traffic (AADT) from short duration tube counts.

In order to establish these seasonal factors, ATR sites are distributed among groups of road sections that demonstrate distinctive seasonal patterns. Experiences of various state Departments of Transportation have shown that the coefficient of variation of monthly traffic volume ranges from about 10% for urban area roads to 25% for the recreational roads. Consequently, the *Traffic Monitoring Guide* recommends using the following groupings of highways: interstate rural, other rural, interstate urban, other urban, and recreational. After all the highways have been divided into the above five groups, ATRs are allocated to each group. The Federal Highway Administration provides no specific rules for distribution of ATR sites. It simply recommends locating five to eight ATR sites in each of the five groups. It also provides the following general guidelines concerning the addition and/or removal of ATR sites:[1]

(1) HPMS sample sections should receive high priority when additional ATRs become available.

(2) New locations for ATRs should be randomly selected from the HPMS sample sections that do not have ATRs currently.

(3) The selection of new ATR locations should be evenly distributed over geographical areas of the state as well as functional classes of highways so as not to favor a particular region or group of highways.

(4) Locations that are currently used by other monitoring program should receive high priority when adding ATR locations.

(5) Existing sites that provide useful traffic information or sites that are "strategically" important should receive high priority when adding ATR sites.

(6) Existing ATR sites with old or malfunctioning equipment should be considered first when removal of ATR sites becomes necessary.

[1] Traffic Monitoring Guide, 1985, Page 3-2-9.

In Minnesota, the grouping of ATR sites differs from the recommendations of the *Traffic Monitoring Guide*. First, the state is divided into two areas: the seven-county metropolitan area and the out-of-state rural area. In the metropolitan area, ATR sites are distributed among interstates, trunk highways, county state aid highways, municipal state aid system roads, and roads leading to the Minneapolis/St. Paul international airport. In rural areas, ATR sites are located on municipal trunk highways, municipal county state aid highways, rural interstates, rural trunk highways, and rural county state aid highways. See Table 2.1.1 for the breakdown and number of ATR sites in each category.

Over the years, a shifting of emphasis on the location of ATRs in Minnesota from one group to another can be seen. In the 1940s, there were sixteen ATR sites installed: fourteen were on rural and metro trunk highways, and two were on rural interstates. By the 1960s, there were about twenty-four ATR sites: twenty of these were on trunk highways, three were on rural interstates, and only one was on metro interstate (I-94 in Washington county.) In the 1970s, metro area interstates started receiving increasing attention, while the number of ATR sites on rural interstates remained relatively unchanged. By the 1980s, the metropolitan area had almost 60% of the ATR sites. Figure 2.1.1 summarizes the number of ATR sites in each category.

According to Mn/DOT, no specific rules are used in deciding how the ATR sites should be distributed. In the past, it appears that new sites were chosen so that they were on or close to an HPMS sample section.

2.2 Highway Performance Monitoring System (HPMS)

HPMS is designed to provide coverage of the road sections that are not monitored continuously by ATRs. "Coverage counts are short duration counts, ranging from 6 hours

Table 2.1.1 Number of ATR sites in each category

Metro Area:	Interstate:	24
	Trunk Highways:	25
	County State Aid Highways	23
	Municipal State Aid System	12
	International Airport	3

		87
Rural Area:	Municipal Trunk Highways	6
	Municipal County State Aid Highways	14
	Rural Interstate	6
	Rural Trunk Highways	30
	Rural County State Aid Highways	8

		64
		Total: 151

to 7 days, distributed throughout the system to provide point-specific information"[2]. Since the monitoring equipment is completely portable and the monitoring period relatively short, the basic issues in the HPMS elements are:

(1) How to schedule tube counting equipment among road sections that do not have ATRs, and

(2) How to convert short duration tube counts to Average Annual Daily Traffic (AADT) estimates.

[2] Traffic Monitoring Guide, page 3-1-1.

Under the Highway Performance Monitoring System described in the *Traffic Monitoring Guide*, all public highways or roads within a state (with the exception of those functionally classified as local) are divided into sections. A section is defined as a segment of a road with uniform attributes. These attributes include pavement type, pavement width, AADT volume, etc.

After roads have been broken down into sections, they are classified into three categories based on their locations: rural, small urban, and urbanized area. Under each category, sections are divided into functional classes. The five functional classes for sections in rural areas are: interstate highways, other principal arterial, minor arterial, major collector, minor collector, and local. For sections in small urban or urbanized areas, the five functional classes are: interstates, other freeways and expressways, other principal arterial, minor arterial, collector, and local. Finally, under each of the resulting classes, roughly thirteen groups are set up using prescribed traffic volumes. This third level of stratification is used to reduce sample size and to insure the inclusion of higher volume sections in the sample. Table 2.2.1 summarizes these three levels of stratification.

After all the public roads in a state are divided into sections and each section is classified, a stratified random sample of road sections is selected by the Federal Highway Administration. These road sections are designated as the "HPMS sample sections" for that state. Currently, Minnesota has about 2,000 HPMS sample sections.

Resources do not permit the continuous monitoring of all sample sections by Automatic Traffic Recorders. For the sample sections that do not have ATRs, the *Traffic Monitoring Guide* recommends a 48-hour tube count on one-third of them each year. Thus, the entire group can be surveyed in a three-year cycle. These 48-hour tube counts

Table 2.2.1 Three Levels of Stratification Used by HPMS

Type of Area	Functional Class	Prescribed Volume Group
Rural	Principal Arterial - Interstate	13 volume groups in each subgroup.
	Other principal Arterial	
	Minor Arterial	
	Major Collector	
	Minor Collector	
	Local	
Small Urban and Urbanized Area	Principal Arterial - Interstate	
	Principle Arterial - Other Freeways and Expressways	
	Other Principal Arterial	
	Minor Arterial	
	Collector	
	Local	

are transformed to AADT estimates using factors such as axle-correction factors, seasonal factors, monthly factors, and growth factors. The AADTs on these HPMS sample sections, whether from ATRs or from estimation, must be reported to the Federal Highway Administration once a year.

Since the *Traffic Monitoring Guide* is an advisory document rather than a Federal standard, each state is free to implement its own traffic monitoring program as long as it reports reliable HPMS data annually to the Federal Government. At the Minnesota Department of Transportation, HPMS is administered by the "Highway Program Department". Traffic-volume data for all the publicly-accessible roads in Minnesota are collected (or estimated) with methods described later in this chapter. These traffic data are stored in Mn/DOT's Traffic Information System database. The Highway Program Department retrieves data from this database, transforms or adjusts traffic-volume data if necessary for designated HPMS sections, and reports them to the Federal Highway Administration.

Even though Mn/DOT uses the HPMS sections for reporting purposes, it does not use them for traffic-volume data collection purposes. Instead, Mn/DOT uses the so-called "road log sections" as the basis for traffic-volume data collection. These road log sections are very similar to the HPMS sections but not necessarily the same. (That is why the Highway Program Department at Mn/DOT may have to adjust traffic-volume data on the HPMS sections for the annual report to the Federal Highway Administration.) Information about the road log sections is stored in the RLG database at Mn/DOT. This database is but one of the many different databases maintained in the Traffic Information System (TIS) at Mn/DOT.

Each road log section is represented in the RLG database as a single record and contains over 100 traffic-related attributes such as route system, surface width, surface type, number of through lanes, etc. Traffic-volume data for each road log section are stored in a separate database and can be retrieved if needed. Because of this set-up, a user can retrieve these road log sections according to any combination of attributes. For example, a user can request traffic-volume data based on route system only. The program will combine consecutive road log sections that are identical in terms of other attributes, and calculate the weighted average traffic-volume for each resulting section. Therefore, the number of sections extracted depends on how the user aggregates the road log sections. The more attributes used, the finer the division. The finest level results if a user specifies all attributes to classify sections, resulting in more than 216,000 sections.

The Minnesota Department of Transportation uses the following procedures to estimate traffic-volume data on these road-log sections:

(1) All the road-log sections are divided into state roads and non-state roads. State roads consist of Interstates, U.S. highways, and Minnesota highways, while non-state roads consist of all other sections. Under each category, sections are classified as either in a metro or rural area. Each road section in the metropolitan area is classified into one of twelve groups resulting from the combination of four monthly traffic-volume trends (Urban, Suburban, Outlying, and Shopping) and three daily traffic-volume trends (Commuter, Mix, and Recreational.) Each out-of-state rural road section is classified into one of three color-coded groups based on its seasonal variation of traffic volumes (Benson, Pisharody, and Yeldan [1986]).

(2) To estimate AADTs for the state road sections, a 48-hour tube count is taken in even years on 7,900 selected sites. These 7,900 sample sites were selected by Mn/DOT to assure "uniform coverage" of the state roads. This set of sites has remained relatively unchanged over the years. The tube counts are transformed into AADT estimates using adjustment factors that have been established for each group described in (1) above. During odd years, the AADT of a particular section is estimated by multiplying the previous year's AADT estimate by the expansion factor established for the group to which that particular section belongs.

(3) To estimate AADTs for non-state road sections in the metro area, approximately 12,400 sites were selected and 48-hour tube counts are taken on a two-year rotating basis (6,200 sites each year.) These sites were subjectively selected by Mn/DOT and have remained relatively unchanged over the years. The sampling issue is more of "when to sample" than "where to sample". Currently, Mn/DOT does not use any statistical procedure in scheduling of tube-count equipment.

(4) For rural non-state roads, approximately 31,800 sites were selected and divided into three groups of 10,600 sections each. In odd years, tube counts are taken from one of the three groups on a rotating basis. In even years, only 3,000 sites are selected for traffic-volume data collection as the tube count equipment must also be used to collect traffic-volume data for state roads. See Table 2.2.2 for a summary of the number of tube counts per year for state and non-state roads.

(5) For all other non-sampled sections, Mn/DOT's personnel use their knowledge about the area to "smooth out" the known AADTs from nearby ATR sites or sampled sites.

Table 2.2.2 Number of Tube Counts per Year

	State Roads		Non-state roads		
Year	Metro	Other	Metro	Other	Total tube counts
1			6,200	10,600	16,800
2	1,500	6,400	6,200	3,000	17,100
3			6,200	10,600	16,800
4	1,500	6,400	6,200	3,000	17,100
5			6,200	10,600	16,800
6	1,500	6,400	6,200	3,000	17,100
Total sites	1,500	6,400	12,400	31,800	

Combining the continuous (ATR) element and the coverage (HPMS) element, it can be seen that there are three levels of accuracy in the AADTs:

(1) AADTs from 151 continuous ATR sites: These are the most accurate.

(2) AADTs from sampled sites: These are AADTs estimated from 48-hour tube counts and adjusted by appropriate expansion factors.

(3) AADTs from interpolation and expert judgment: These are AADTs of the sections that never have any kind of counts taken on them.

2.3 Special Needs Element

Special needs elements are designed to supplement the ATR and the HPMS elements. They exist to satisfy the specific traffic-data needs of individual states. According to the *Traffic Monitoring Guide*, the purpose of the special needs element is "to provide wide flexibility, to encompass the diversity of situations, and to allow each State to design its program in accordance with its self-defined needs and priorities" (p. 3-4-2)

The special needs elements can be divided into two major categories: system needs and point-specific needs. Some of the most important system needs are the periodic development of volume-flow maps, the determination of volume-group strata for the HPMS, and the development of subunit Vehicle Miles Traveled (VMT) estimates. Point-specific needs include crucial traffic information for highway projects, probably the most important concern from the point of view of state management of highway programs.

2.4 Why the Current Procedures Are Inadequate

The three traffic-monitoring elements--as designed by the U.S. Department of Transportation and implemented by Mn/DOT--are inadequate in the following areas:

(1) They ignore "use profiles" of road sections and this results in inefficient sampling. For example, if traffic volumes on two road sections are perfectly correlated, monitoring one of the two sections gives exactly the same information as monitoring both sections. In this case, it would be a waste of resources to place traffic-monitoring equipment on both sections. Although the HPMS sample sections (selected through stratified random sampling) provide "uniform coverage" of the public roads, they do not provide a mechanism to explicitly exclude road sections of similar or identical profiles. Thus, two road sections that have highly correlated traffic volumes but are far apart could be selected at the same time under the current procedure.

(2) Estimation of most AADTs is judgmental. Road sections monitored by ATRs or portable tube count equipment account for only a small percentage of all the road sections in a state (in Minnesota it is less than 8%). The majority of AADT estimates are judgments made by traffic engineers. This procedure could result in different estimates by different personnel.

(3) It is impossible to develop statistical measures of precision for the AADT estimates when they are judgmental. (Of course, a traffic engineer could provide subjective measures of precision and confidence. For a discussion of the cognitive processes involved in constructing such measures, see Smith, Benson, and Curley [1991].)

In this research, we develop a number of statistical methods that eliminate or alleviate these problems.

2.5 Scope of This Report

In this report, we focus on the ATR element rather than the HPMS or Special needs elements for the following reasons:

(1) The Special needs element is too diverse and too state-specific to be covered in this research. We are interested in developing general sampling techniques that can be applied in any state.

(2) The sampling problems associated with the HPMS element are more complex because they involve not only where to sample but when to sample. The methodology we have developed for the ATR element is applicable to the sampling problem of the HPMS element, but would need to be extended to account for seasonal fluctuations.

The ATR element is the backbone of any traffic monitoring program. Even if ATRs are eventually phased out by more sophisticated equipment (such as the "Video Detection System" developed by Professor Michalopolus of the University of Minnesota), the question of where to locate such equipment remains.

In summary, the question addressed in this research is: How can ATR sites be optimally selected from all potential locations to estimate AADTs as accurately and efficiently as possible? Since the problem in simplest term is one of sample-selection, a review of sampling techniques is presented in the next chapter.

CHAPTER 3
REVIEW OF SAMPLING METHODOLOGIES

In this chapter, we discuss the development of the finite population sampling methodology and two competing concepts in the theory of inference for finite population sampling: design-based and model-based sampling. In succeeding chapters, we use model-based sampling to develop algorithms for optimally locating ATR sites.

3.1 Finite Population Sampling Methodology

Researchers have focused on two issues related to finite population sampling: sample design and inference. Smith [1976] provided an excellent review of the relationship between these two issues. He stated:

> At the end of the nineteenth century, statistical inference and survey design were proceeding along separate paths. There was a well-established theory of inference for large samples from infinite populations based on the Central Limit Theorem. The main method of inference was Bayes's theorem, but since the samples were large the posterior distributions were virtually independent of the prior and so were determined primarily from the sample data.
>
> Survey design was in its infancy, and there was an extensive debate, under the heading of "the representative method", about the scientific validity of any form of sampling from finite populations. To some only censuses were allowable (p. 183).

According to Smith [1976], inference from finite population sampling was gradually recognized as a valid survey method after Kiaer [1901] demonstrated empirically that stratified samples could provide good estimates of finite population totals and means.

In 1903, the International Statistical Institute recommended the adoption of stratified sampling with proportional allocation as an acceptable scientific method of data collection.

Bowley [1906] brought together survey sampling and inference. Bowley developed a method to evaluate the accuracy of estimates from large random samples selected from a large finite population and established the computation of variance for a stratified random sample. He also proposed balanced sampling when the groups were of different sizes and the sample mean of a random sample differed markedly from the population mean. (In balanced sampling, a purposive sample is chosen such that the sample mean is equal to the population mean for some control variables.)

3.2 The Neyman Revolution

Bowley's contribution inspired Neyman to establish a new framework for inference in finite population sampling. In his influential 1934 paper, Neyman laid down a basis for the logic of inference based on the confidence interval argument. This confidence interval statement is actually a frequency statement referring to the proportion of confidence intervals derived from all possible samples that contain the parameter of interest in the population. In this approach, it is not necessary to make any assumptions about the distribution of the variable of interest in the population because randomization in sample design provides the needed probability structure for inference. The probability distribution of the variable of interest generated by random sampling is called the p-distribution and is the fundamental element in the subsequent development for inference in finite population sampling. This approach to sampling theory has been described as *design-based sampling*. (See Cochran [1977] for a complete discussion of this methodology.)

Neyman's confidence interval approach and lack of assumptions about the population were regarded as being too general by several statisticians including Bowley and R.A. Fisher. Another problem in Neyman's framework is the inability to make predictions about the points that are not sampled. If a finite population has N points and a sample of size n is chosen, we know everything about values in the sample but nothing about the values not in the sample. The only way to obtain information on the $(N - n)$ unobserved values is to relate them by a mathematical model to the n observed values. One possible solution is the "superpopulation" approach. This approach is sometimes called the *model-based sampling* or the *prediction* approach. Differences between this approach and the more conventional *design-based sampling* approach will be discussed in the next section.

3.3 Differences Between Design-based and Model-based Sampling

In design-based sampling, the parameter of the finite population under investigation is treated as having a fixed but unknown value. A sample of size n is chosen from the population and an estimate is derived using the probability structure generated by the Central Limit Theorem. In the superpopulation approach, however, the finite population $\{y_1, y_2,..., y_N\}$is assumed to be a realization of random variables $\{Y_1, Y_2,..., Y_N\}$. As a result, the parameter of the finite population is treated as a random variable. The joint distribution of $Y_1, Y_2,..., Y_N$ is usually denoted by ξ. The set of conditions and assumptions that describe ξ, usually specified by the sampler, is called a "superpopulation model". This superpopulation model can be used to describe the process that generates the finite population. For example, let $y_1, y_2,..., y_N$ denote the AADTs of the N road sections in the state of Minnesota in 1991. These N AADTs can be thought of as just a random realization (in 1991) of an "AADT-generating" process that generates AADTs year after year according to the joint distribution of $Y_1, Y_2,..., Y_N$, denoted by ξ. To specify ξ, knowledge about the superpopulation is needed. This knowledge is usually obtained through the use of "auxiliary variables". For instance, an auditor may specify ξ to be a simple linear regression model with audit value and book value as the dependent and independent variable, respectively. In this case, the audit value is the primary variable and the book value is the auxiliary variable. After the regression model is constructed, a sample of n accounts can be selected by the auditor (purposely) to optimize a pre-specified criterion related to the model used. Audit values of the remaining (N - n) accounts not included in the sample can be estimated via the regression model. (See Ko [1986] for a detailed discussion of this subject.) This methodology is usually referred to as *model-based sampling.*

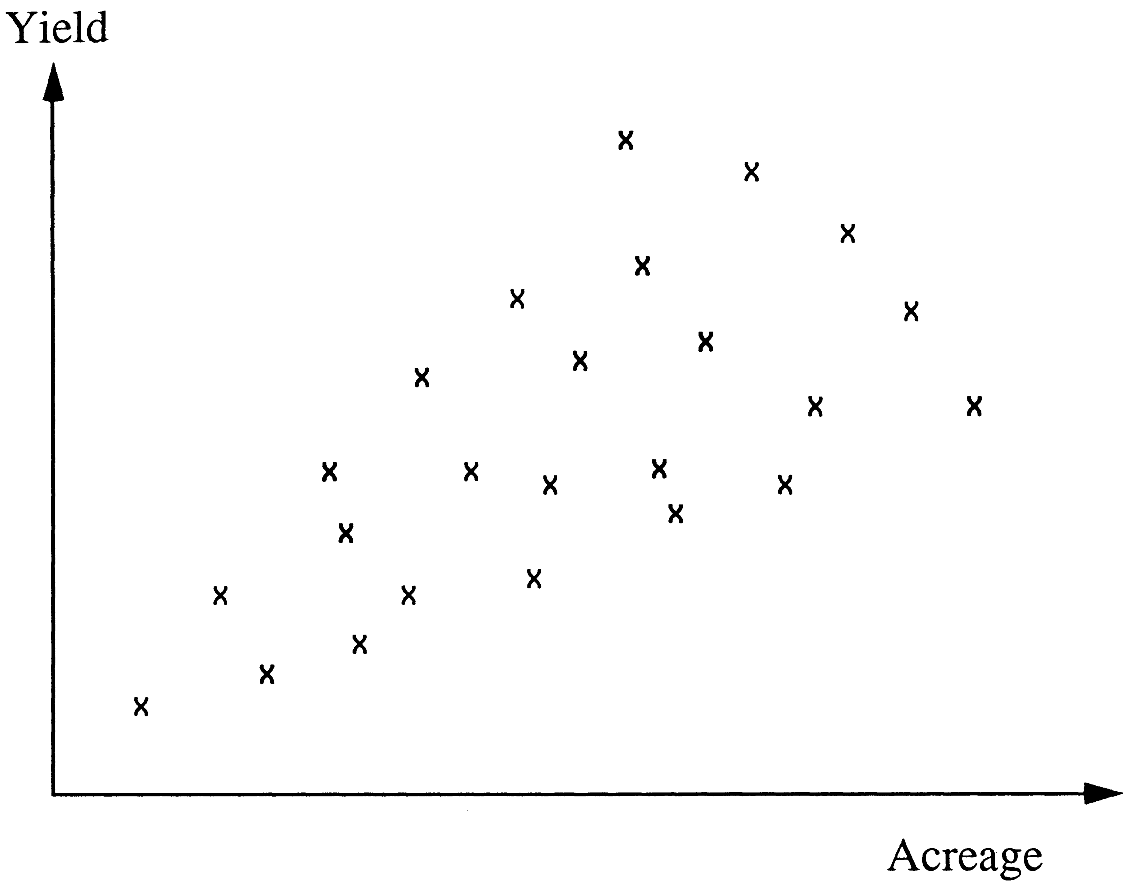
To further illustrate the difference between *design-based sampling* and *model-based sampling*, consider the problem where we are to estimate the average yield of corn fields ($\overline{Y}$) in a particular county by selecting n corn fields from a total of N fields. Suppose from past experience, we know that the yield of a field (y_i) is proportional to the acreage of the field (x_i) and that the relationship between these two variables can be described in the scatter plot in Figure 3.3.1.

The naive design-based approach to this estimation problem is to ignore the information in the scatter plot, select a random sample of corn fields, and use the mean yield from this sample ($\overline{y}$) as an estimate of the population average. A better design-based approach is to select a random sample and use the ratio estimator

$$\hat{\overline{Y}}_R = \frac{y}{x}\overline{X}$$

where $\overline{X}$ is the mean acreage of corn fields in the population and x and y are the sample totals of the x_i and y_i, respectively . This approach is more precise than the naive approach since it utilizes additional information about the acreage of corn fields.

Figure 3.3.1 Relationship Between Acreage and Yield of Corn Fields



A model-based sampler, however, will most likely model the relationship between yields and acreages as a straight line through the origin and assume that the error variances are proportional to acreages of corn fields. From past data, the sampler can construct a simple linear regression model using acreage as a predictor of yields. Based on this regression model, a purposive sample is selected and a special estimator (discussed in the next section) is used to estimate the population mean. It has been shown that the purposive sample under the model assumptions in this example is made up of the n largest corn fields. A sample so selected will minimize the mean squared error of predicting $\overline{Y}$ because by including largest corn fields in the sample, we eliminate the need to estimate the yields of these corn fields that have the largest error variances.

Another difference between the design-based and model-based approaches concerns the prediction of the individual yields of corn fields. In the design-based approach, we know nothing about the yields of corn fields not included in the sample. In model-based sampling, these predictions can be made through the regression model. This is why model-based sampling is sometimes called the "prediction" approach.

3.4 Royall's Prediction Approach

Model-based sampling received increased attention after Royall published his controversial paper in 1970. This paper deals with estimating population totals. In Royall's model, the population of interest consists of N identifiable units $S=\{1,..., N\}$. Associated with each unit i are two quantifiers x_i and y_i, with x_i known and y_i fixed but unknown. The relationship between x_i and y_i can be described roughly as a straight line passing through the origin while the scatter of (x_i, y_i) about the line increases with larger x. A subset s of size n is to be selected from $\mathbf{S}$, and the y values of the sampled units observed. The purpose is to estimate

$$T = \sum_{i=1}^{N} y_i \,.$$

Following conventional terminology, we shall call the combination of a sampling design and an estimator a *strategy*. The general notation for this will be (s,t). For example, a strategy of (*srs*, *sample mean*) involves using simple random sampling without replacement and the sample mean as an estimator. Many strategies may be used in estimating T. Royall used two familiar strategies in his paper to demonstrate his arguments. The two strategies are: (pps, $\hat{T}_{HT}$), and (srs, $ratio$). In the first strategy, a random sample is selected so that the probability of including a unit is proportional to the unit's value x_i (pps - means probability proportional to size). T is then estimated by the Horvitz-Thompson (1952) estimator:

$$\hat{T}_{HT} = \frac{1}{n}\sum_{s}\left(\frac{y_i}{x_i}\right)\sum_{i=1}^{N} x_i .$$

In the second strategy, a simple random sample (srs) is selected and T is estimated by the ratio estimator

$$\hat{T} = \left(\frac{\sum_{s} y_i}{\sum_{s} x_i}\right)\sum_{i=1}^{N} x_i .$$

These two strategies were studied under the following model: $y_1,\ldots,y_N$ are considered to be realized values of $Y_1,\ldots,Y_N$ which have the properties:

$$\begin{aligned} &E(y_i) = \beta x_i \\ &Var(y_i) = \sigma^2 v(x_i) \\ &Cov(y_i, y_j) = 0 \qquad (i, j = 1,\ldots,N; i \neq j) \end{aligned}.$$

The function $v(x_i)$ is known; the constants β and σ^2 are unknown, and ξ denotes the joint probability distribution of $Y_1,\ldots,Y_N$.

A strategy $(p,\hat{T})$ will be considered better than another strategy $(p',\hat{T}')$ if the following is true:

$$MSE(p,\hat{T}) < MSE(p',\hat{T}'),$$

where mean squared error of $(p,\hat{T})$ is defined as:

$$MSE\left(p,\hat{T}\right) = E_{\xi}\left\{\sum_{S} p(s)(\hat{T}-T)^2\right\}$$

and $p(s)$ stands for the probability of selecting sample s.

An estimator $\hat{T}$ will be called p-unbiased (design-unbiased) with respect to a sampling plan p if

$$\sum_{S} p(s)\hat{T} = T$$

and will be called ξ-unbiased (model-unbiased) if

$$E_{\xi}(\hat{T}-T) = 0.$$

An estimator can be p-unbiased but not ξ-unbiased and vice-versa. For example, if p is a simple random sampling plan, then N times the sample mean is a p-unbiased estimator but not a ξ-unbiased estimator. On the other hand, under Royall's model above, the ratio estimator can be shown to be an ξ-unbiased estimator but not p-unbiased.

By the Gauss-Markov theorem, the estimator which minimizes the mean squared error in Royall's model (regression through the origin) has the following structure:

$$\hat{T}^* = \sum_{s} y_i + \hat{\beta}^* \sum_{s'} x_i,$$

where

$$\hat{\beta}^* = \frac{\sum_{s} \frac{x_i y_i}{v(x_i)}}{\sum_{s} \frac{x_i}{v(x_i)}}.$$

Royall proved that for any sampling plan p, if $\hat{T}$ is any linear estimator that is either ξ-unbiased or whose mean squared error is a bounded function of β, then $MSE(p,\hat{T}^*) \leq$

$MSE(p,\hat{T})$. In the cases where $v(x)=1$, x, and x^2, best linear ξ-unbiased estimators for T were shown by Royall to be:

$$\hat{T}_0 = \sum_s y_i + \left(\frac{\sum_s x_i y_i}{\sum_s x_i^2} \right) \sum_{s'} x_i ,$$

$$\hat{T}_1 = \sum_s y_i + \left(\frac{\sum_s y_i}{\sum_s x_i} \right) \sum_{s'} x_i ,$$

and

$$\hat{T}_2 = \sum_s y_i + \left[\frac{1}{n} \sum_s \left(\frac{y_i}{x_i} \right) \right] \sum_{s'} x_i ,$$

respectively.

Royall showed that if (1) $\max_i \ (nx_i) \leq \sum_{j=1}^{N} x_j$ and (2) $\frac{v(x)}{x^2}$ is a nonincreasing function, then for any sampling plan with fixed size n,

$$MSE(p,\hat{T}_{HT}) \geq MSE(p,\hat{T}_2).$$

That is, the optimal estimator $\hat{T}^*$ should be preferred over the Horvitz-Thompson estimator given any sampling plan.

After proving $\hat{T}^*$ to be the optimal estimator for all p under the given model with variance function $v(x)$, Royall turned his attention to the choice of sampling plan. He showed that if $v(x)$ is nondecreasing and $v(x)/x^2$ is nonincreasing, then $MSE(p^*,\hat{T}^*) \leq MSE(p,\hat{T})$ for any sampling plan p and any ξ-unbiased estimator $\hat{T}$. In the above expression, p^* is the sampling plan which always selects the n units having the largest x values. Therefore, when $v(x) = x$, the conventionally preferred strategy $(p_{srs}, \hat{T}_1)$ is dominated by $(p^*, \hat{T}_1)$ in which the ratio estimator is used with the purposive sampling plan. This should not be surprising since under this particular model, p^* selects the n units which have the largest x values (and therefore the largest variances). This sampling plan leaves only the units with the smallest variances to be estimated, thereby minimizes the mean squared error of prediction. Furthermore, p^* also generates the best estimate of the slope of the regression line which in turn makes the ratio estimator more reliable.

3.5 Model Robustness

Neyman pointed out that since the optimality results of the model-based approach depend heavily on the assumption that the true form of the superpopulation model is known to the sampler prior to sampling, it is dangerous to draw a sample based on an unverified model. A misspecified model could lead to potentially serious bias in the estimate. Royall and Herson [1973] discussed two possible solutions for model uncertainty: balanced sampling and stratification on a size variable.

In balanced sampling, the sampler deliberately selects a sample such that the sample moments of a control variable match its population moments. If a sample is balanced through the Jth moment, then the ratio estimator simplifies to the expansion estimator and is the best linear unbiased estimator for any Jth degree polynomial regression model in which the variance function is proportional to x^j for some j between 0 and J. Scott, Brewer and Ho [1978] extended their results to include more general variance functions and regression estimators. Later, Royall and Cumberland [1981] suggested a new type of balanced sampling design: choose the sample that best matches the sample cumulative distribution function to the population cumulative distribution function.

There are at least two drawbacks to using balanced sampling, however. The first is the loss of efficiency, particularly if the ratio of the maximum x-value in the population to the minimum x-value in the population is greater than 5. Royall compared the trade-off between model robustness and efficiency to that of deciding how much insurance to buy -- something that samplers must decide by themselves. A second difficulty is that the exact match of moments is usually impossible. Royall and Herson suggested that as long as J and the sampling fraction n/N are both small, an approximate match could be accomplished. The ratio estimator is "approximately unbiased" in "approximately balanced" samples.

Another suggestion they made to overcome model uncertainty was to stratify the entire population into H strata so that the first stratum has the N_1 units with the smallest x values, the second stratum has the next N_2 smallest units, and so on. The optimal stratification is to choose N_h for h=1,.., H so that $\sum_{1}^{H} N_h\sqrt{x_h}$ is minimized. Balanced sampling and the ratio estimator were then used within each stratum. They showed that this stratified balanced sampling strategy is more efficient than the simple balanced sampling strategy when allocation of observations to strata is made in the optimal manner. The optimal allocation is achieved when n_h, the number of units sampled in stratum h, is proportional to $N_h\sqrt{x_h}$. Despite all these efforts, Tallis [1986] argued that balanced sampling is the optimal sampling design only if there is homoscedasticity.

3.6 Other Optimality Criteria

Minimizing the mean squared error of the best linear unbiased estimator of the population total obviously is not the only optimality criterion that can be applied in model-based sampling. There are situations where we are more interested in predicting individual

y_i than the population total $\sum_{1}^{N} y_i$. The minimum variance linear unbiased estimator in this case is simply $\hat{y}_i$, the weighted least squares estimate of $E(y_i)$, and a suitable optimality criterion here is the minimax approach.

In Wynn [1977a, b], the minimax approach was used with model-based sampling. This approach attempts to minimize the maximum normalized mean squared error of prediction over the non-sampled units. Wynn has established a necessary condition for a sample to be a minimax sample: in the homoscedastic case, the minimax MSE of $\hat{y}_i$ is bounded above by $\sigma^2(n+1)/(n-k+1)$ where σ^2 is the error variance, n is the sample size, and k is the number of predictors in the model. This upper bound can be used to check the admissibility of sampling designs but itself does not guarantee the minimax sample, i.e., there could be more than one sample which satisfies this condition. In practice, numerical methods are needed to find the nearly minimax samples.

3.7 Application of Model-based Sampling in Practice

A number of researches have shown that the concept of model-based sampling can be applied to real-world problems. Royall [1973] used model-based sampling in hospital discharge surveys. More recently, Godfrey, Roshwalb, and Wright [1984] compared model-based stratification with two conventional stratification techniques in estimating inventory cost and concluded that model-based approach is superior under a wide range of model parameters. Karmel and Jain [1987] compared MSE efficiency between model-based and conventional strategies in estimating capital expenditure and concluded that ratio estimation, optimal allocation, and purposive sampling is much more efficient than design-unbiased strategies such as random sampling.

Accounting is another area in which model-based sampling can be applied. In accounting, statistical sampling methods provide quantifiable measures of the auditor's risks concerning judgments about the magnitude of accounting errors. Currently, the statistical sampling methods used in auditing rely heavily on design-based sampling methodologies. Random selection of the sample provides the necessary probability distribution for the statistical inferences to be made about the population parameters. Recently, model-based sampling has been advocated by several auditors for use in accounting . Motivation and a much more detailed discussion of the model-based sampling approach in accounting can be found in Ko [1986]. Later, Ko, Nachtsheim, Duke, and Bailey [1988] used a simulation study to measure the robustness of the various model-based approaches to changes in assumptions about the target population. Their simulation results suggested that the form of the variance function $v(x)$ is a more critical determinant of performance than the form of regression model. They concluded that while substantive gains in efficiency are possible through model-based sampling, randomization-based strategies should be preferred in the absence of reliable prior information about the assumed form of the variance function.

CHAPTER 4
IDENTIFYING A SUPERPOPULATION MODEL: AN EMPIRICAL APPROACH

4.1 Introduction and Summary

Model-based sampling is a viable alternative to design-based strategies only if a superpopulation model can be identified. In this chapter we summarize an exploratory regression analysis that led to the identification of one very intuitive and statistically supportable superpopulation model that can be used for predicting AADTs. We also discuss the variance structure of this regression model. In the last section, we discuss the problem of spatial correlation among population units and its ramifications for AADT estimation.

The regression analysis indicates that the four most important factors affecting AADTs are: county population, number of through lanes on the roadway, whether the section is on a state road or non-state road, and whether it is in a rural area or an urban area. It was not surprising to find that AADT is highly correlated with county population size since one would expect higher traffic volumes in highly populated areas. The high correlation between AADT and number of through lanes has at least two possible explanations: (1) traffic engineers foresee the demand and design the roadway accordingly or (2) the roadways with more through lanes have faster traffic flow and in turn "attract" more traffic from competing routes. The other two predictors are indicator variables and they coincide somewhat with the variables that Minnesota Department of Transportation uses to stratify road sections. The following is a more detailed description of the regression analysis.

4.2 Regression Analysis

4.2.1 Data Collection and Screening

In order to construct a regression model to describe the "superpopulation," a sample with reliable traffic data must be obtained. One obvious choice is to use traffic data from existing ATR sites. We used the road-log database (described in Chapter 2) in the Traffic Information System at Mn/DOT. This database contains over 216,000 records. Each record contains detailed attributes of a road segment. With the generous help of Mn/DOT's Karl Olmstead[1] and Dennis Carroll, we were able to retrieve the 1988 records for 134 of the 151 ATR sites. Due to problems such as equipment failure and power outages, not all sites have complete traffic-volume data. After initial screening, 122 ATR stations were used to develop the regression model.

4.2.2 Selection of Predictors

The potentially useful predictors of traffic volume were chosen from variables currently available in the road-log (RLG) database. There are 106 data items associated with each record in the database. These data items provide detailed information of a road section such as its route system, intersection category, surface width, surface type, number of traffic lanes, and so on. Since the database is designed to include as many road-sections attributes as possible, not all records have the entire set of 106 data items.

We considered the following thirteen data items in our regression analysis:

(1) ROUTE-SYSTEM: This variable identifies the ownership of the road section. Interstate trunk highways, U.S. trunk highways, and Minnesota trunk highways

[1] No longer with MN/DOT

are considered state-owned roads. The other seventeen route systems such as county state-aid highways, county roads, township roads, and so on are considered non-state roads.

(2) POP-FROM-CITY: This variable indicates the population size of the city where the road section is located.

(3) POP-FROM-CNTY: This variable indicates the population size of the county where the road section is located,

(4) RUR-URB-FROM-CITY: This is the rural/urban designation code for the nearest city. A city is considered rural if its population size is smaller than 5,000; urban if its population size is between 5,000 and 49,999; urbanized if its population size is 50,000 or greater.

(5) FUNCT-CLASS: This code identifies the usage of a road section. There are six and eight functional classes for rural and urban road sections, respectively. The six functional classes for rural sections are: interstate, other principal arterial, minor arterial, major collector, minor collector, and local roads. The eight functional classes for urban sections are: interstate, other connecting freeway, other non-connecting freeway, other connecting link, other non-connecting link, minor arterial, collector, and local roads.

(6) INTERSECT-CATEGORY: This code indicates the route system of any intersecting road sections.

(7) SPECIAL-SYSTEMS: This code indicates whether a road section has a special status such as "national forest highway" or "great river road".

(8) FED-AID-SYS: This code indicates whether a road section receives federal aid, and if yes, what type of federal aid.

(9) CONTROL-OF-ACCESS: This code indicates whether access to a road section is fully controlled, partially controlled, or not controlled.

(10) TOTAL-THRU-LANES: This is the total number of through lanes (in both directions) on the road section.

(11) TRUCK-ROUTE-CLASS: This code identifies the type of truck-route. There are eight truck-route categories.

(12) SURF-WID: This number identifies the width of the road sections (in feet) including sidewalks (if any) and non-traffic-carrying lanes (such as space for parking).

(13) SURF-TYPE: This code identifies the type of surface on a road section. There are twenty-five categories.

After retrieving these data items for existing ATR sites, we dropped the variables that were not usable (see next section) and used the all possible regressions routine and other standard subset selection methods to reduce the number of predictors to four: route system, population size of the county (where the section is located), total number of through lanes, and rural/urban identification code (with combined group for urban and urbanized areas). We also considered higher-order terms for the continuous variables. The final model can be summarized as follows:

$$AA\hat{D}T = \hat{\beta}_0 + \hat{\beta}_1 X_1 + \hat{\beta}_2 X_2 + \hat{\beta}_3 X_1 X_2 + \hat{\beta}_4 X_3 + \hat{\beta}_5 X_4 + \hat{\beta}_6 X_2^2$$

where X_1 = county population size,

X_2 = total number of through lanes,

X_3 = state/nonstate code,

and X_4 = rural/urban code.

The regression results in Table 4.2.1 indicate a good fit ($R^2 = 86.7\%$) and significant predictors (all but one predictor have p-values < 0.0001). From the regression equation, we can also see that state roads have higher AADTs than nonstate roads and rural

Table 4.2.1 Regression Model using Centered and Standardized Data (AADT's not Transformed)

```
The regression equation is
AADT = - 0.0636 - 0.510 POP. - 0.569 LANE + 1.06 POP*LANE + 0.332
       STATE=1 - 0.252 RURAL=1 + 0.888 LANE^2

Predictor        Coef        Stdev     t-ratio        p        VIF
Constant     -0.06361      0.06499       -0.98    0.330
POP.         -0.50958      0.08893       -5.73    0.000        6.9
LANE          -0.5693       0.1459       -3.90    0.000       18.5
POP*LANE      1.05696      0.09477       11.15    0.000        7.8
STATE=1       0.33160      0.08597        3.86    0.000        1.6
RURAL=1      -0.25180      0.09378       -2.68    0.008        1.9
LANE^2         0.8883       0.1291        6.88    0.000       14.4

s = 0.3736       R-sq = 86.7%      R-sq(adj) = 86.0%

Analysis of Variance

SOURCE         DF          SS           MS         F         p
Regression      6     104.946       17.491    125.29     0.000
Error         115      16.054        0.140
Total         121     121.000


SOURCE         DF      SEQ SS
POP.            1      48.931
LANE            1      35.630
POP*LANE        1      12.903
STATE=1         1       0.640
RURAL=1         1       0.234
LANE^2          1       6.608
```

roads have lower AADTs than roads in urban areas. The relationships between AADT and county population size and number of traffic lanes are less clear because of the second order terms involved. The residual plot and normal probability plot of this regression model are presented in Figure 4.2.1 and Figure 4.2.2, respectively. The residual plot clearly indicates the violation of the constant error variance assumption of the regression model and the normal probability plot also indicates possible violation of the normality assumption.

Figure 4.2.1 Residual Plot for the Original Regression Model (AADT's not Transformed)

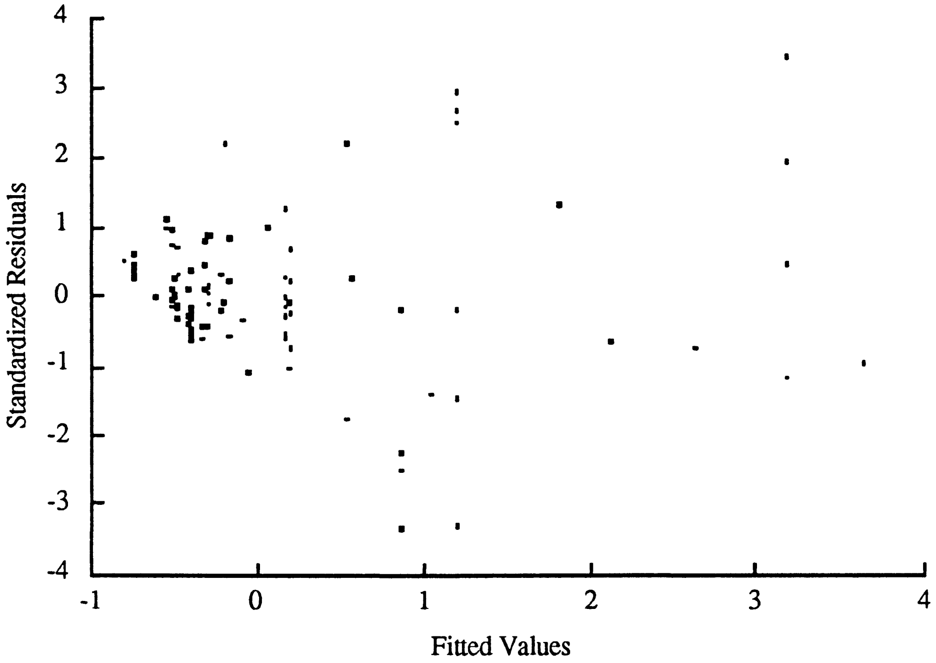
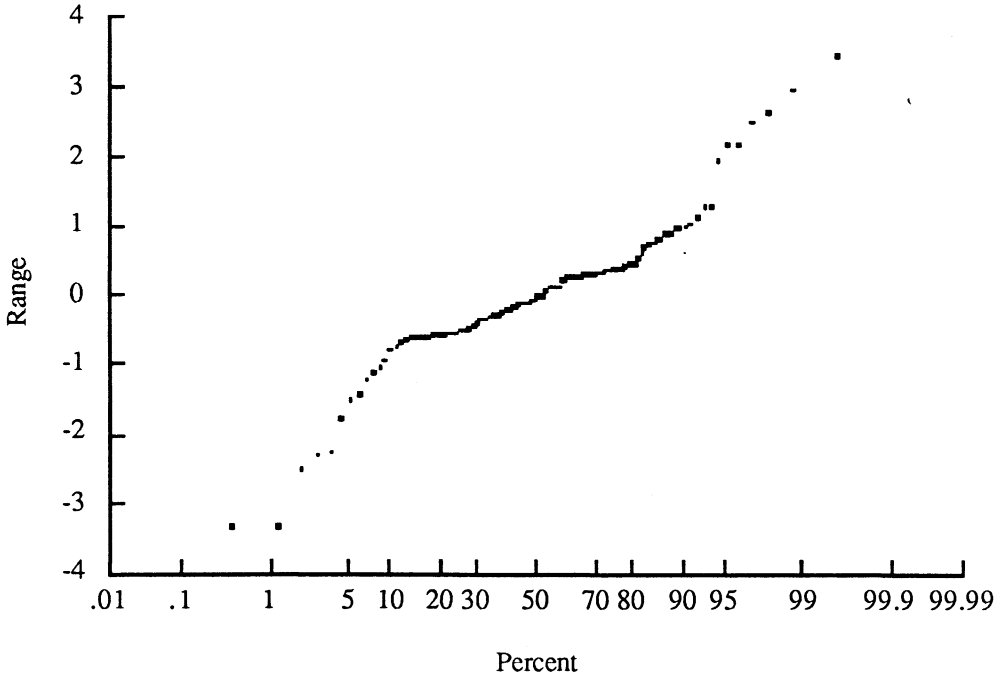


Figure 4.2.2 Normal Probability Plot for the Original Regression Model (AADT's not Transformed)



4.2.3 Limitations of the Regression Analysis

Of the thirteen data items initially selected for the regression analysis, some of them were not usable. For example, city population size is missing for the majority of the road sections. Another limitation is that potentially important predictors of traffic-volumes were not available. For example, the population size within certain distances of a road section should be a better predictor (than the county population size). If the population size within a given radius were available, we believe that further improvements in the regression model could lead to more accurate estimation of AADTs. Other predictors that are potentially useful include: major intersections within a given radius, peak-hour volume, and geographic location of a road section.

4.2.4 Transforming the Response Variable

Because of the violation of the constant variance and normality assumptions of the ordinary least squares regression model, a Box-Cox transformation was performed on the response variable AADT. The results indicated that the optimal power transformation λ^* occurred at $\lambda^* \approx 0.2$. Results from the transformed model did conform to the constant variance and normality assumptions (Table 4.2.2, Figures 4.2.3 and 4.2.4.) A log transformation ($\lambda' = 0$) was also considered; its results are summarized in Table 4.2.3, Figure 4.2.5, and Figure 4.2.6. Comparing the residual plots of the log transformation model with those of the optimal transformation model, one can see that log transformation also satisfies the constant variance and normality assumptions. Cook and Weisburg's [1983] score test failed to detect any non-constant variance violation in the log transformation model.

Table 4.2.2 Regression Model after Box-Cox Transformation

```
The regression equation is
ADT02 = 6.25 + 0.483 POP. + 1.73 LANE + 0.880 STATE=1 - 1.17 RURAL=1
        - 0.816 LANE^2

Predictor        Coef       Stdev     t-ratio        p        VIF
Constant       6.2506      0.1296       48.23    0.000
POP.          0.48328     0.08767        5.51    0.000        1.7
LANE           1.7326      0.2598        6.67    0.000       14.7
STATE=1        0.8798      0.1715        5.13    0.000        1.6
RURAL=1       -1.1687      0.1871       -6.25    0.000        1.9
LANE^2        -0.8164      0.2382       -3.43    0.001       12.3

s = 0.7458       R-sq = 85.9%      R-sq(adj) = 85.3%

Analysis of Variance

SOURCE        DF          SS          MS          F        p
Regression     5     394.610      78.922     141.89    0.000
Error        116      64.522       0.556
Total        121     459.133


SOURCE        DF      SEQ SS
POP.           1     191.171
LANE           1     160.187
STATE=1        1       8.347
RURAL=1        1      28.373
LANE^2         1       6.532
```

The results of these transformations demonstrate that there is indeed a regression model that satisfies the necessary assumptions. However, since our objective is precise estimation in the original scale, in what follows we focus on the original regression model. The basic model assumptions are as follows:

$$E(y_i) = f(x_i)^T \beta$$

$$Var(y_i) = \sigma^2 v(x_i)$$

$$Cov(y_i, y_j) = 0 \qquad (i, j = 1, \ldots, N; i \neq j).$$

Two issues related to this regression model need to be addressed:

(1) Since the original regression model exhibits non-constant variance, the variance function, $v(x_i)$, needs to be identified. This issue is discussed in the next section.

(2) The regression model assumes uncorrelated errors. This assumption may not be appropriate because of potential spatial correlation in the sample. This issue is discussed in section 4.4.

Figure 4.2.3 **Residual Plot for the Regression Model after Box-Cox Transformation**

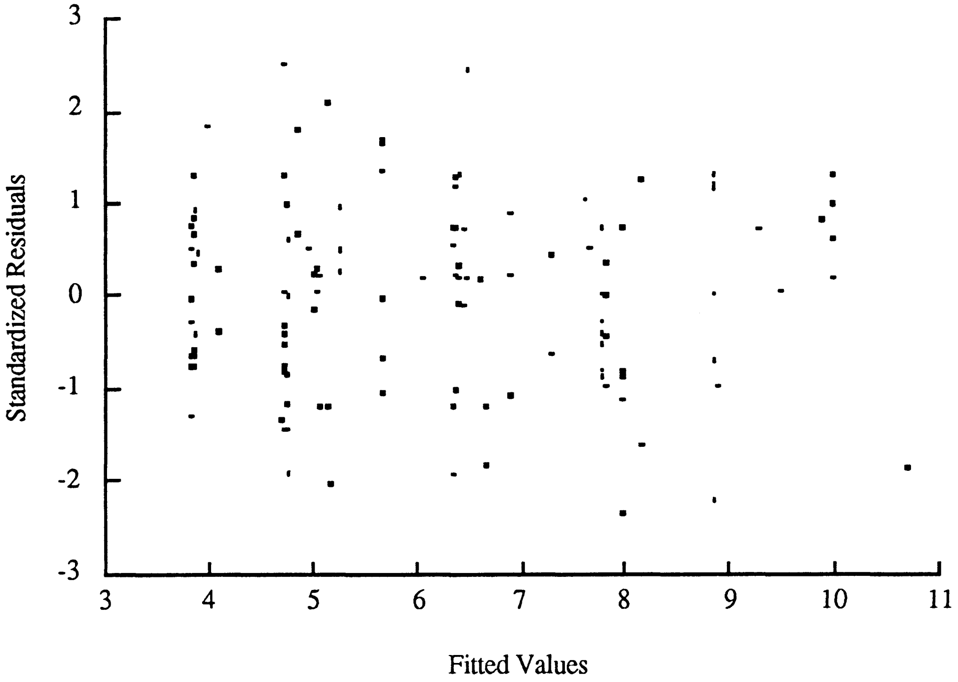


Figure 4.2.4 **Normal Probability Plot for the Regression Model after Box-Cox Transformation**

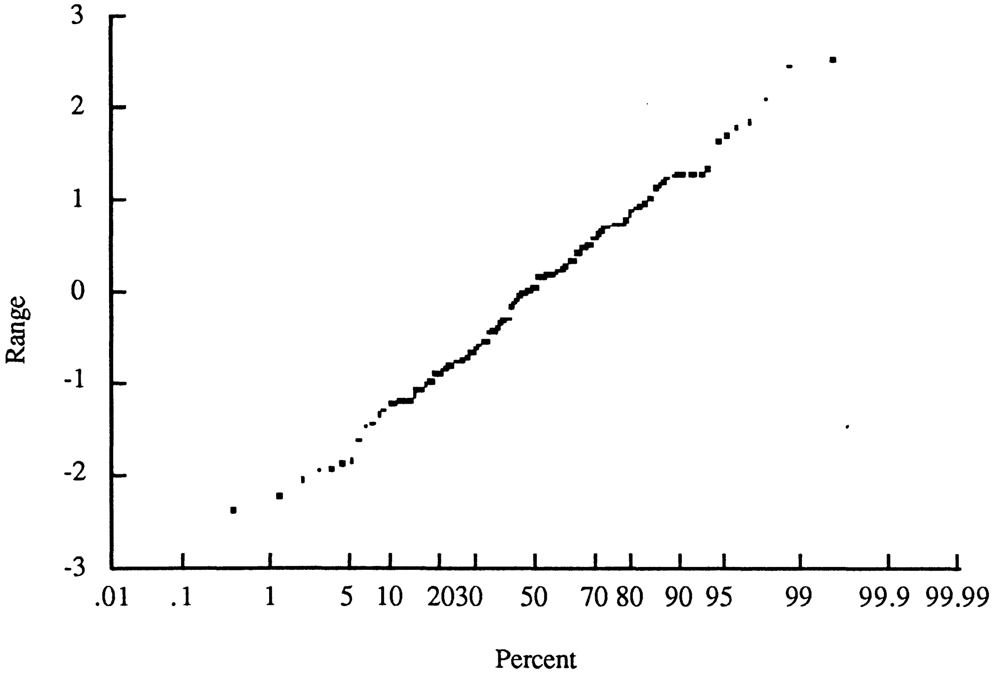


Table 4.2.3 Regression Results of Log Transformation Model

```
The regression equation is
LOGADT = - 0.0000 + 0.725 POP. + 0.977 LANE + 0.232 STATE=1 - 0.277
         RURAL=1 - 0.492 POP^2 - 0.559 LANE^2

Predictor       Coef       Stdev    t-ratio        p       VIF
Constant    -0.00000     0.03803      -0.00    1.000
POP.          0.7254      0.2216       3.27    0.001      33.7
LANE          0.9769      0.1465       6.67    0.000      14.7
STATE=1      0.23199     0.04771       4.86    0.000       1.6
RURAL=1     -0.27669     0.06043      -4.58    0.000       2.5
POP^2        -0.4918      0.2029      -2.42    0.017      28.2
LANE^2       -0.5594      0.1343      -4.16    0.000      12.4

s = 0.4200      R-sq = 83.2%     R-sq(adj) = 82.4%

Analysis of Variance

SOURCE       DF          SS          MS         F        p
Regression    6     100.711      16.785     95.14    0.000
Error       115      20.289       0.176
Total       121     121.000

Continue?
SOURCE       DF      SEQ SS
POP.          1      44.859
LANE          1      38.825
STATE=1       1       2.371
RURAL=1       1      10.382
POP^2         1       1.215
LANE^2        1       3.060
```

Figure 4.2.5 Residual Plot of the Log Transformation Model

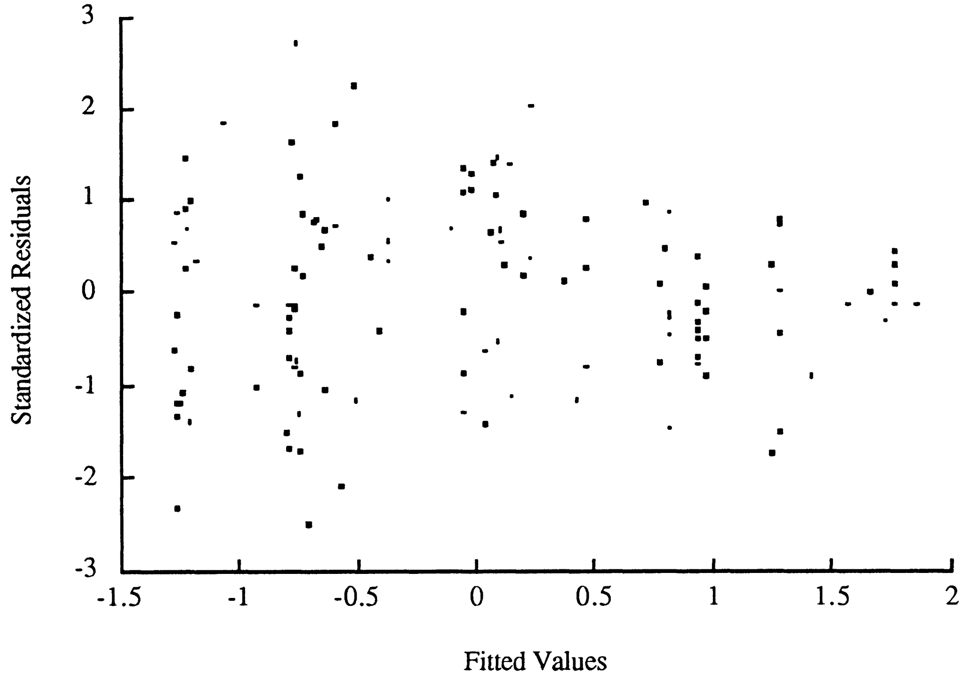
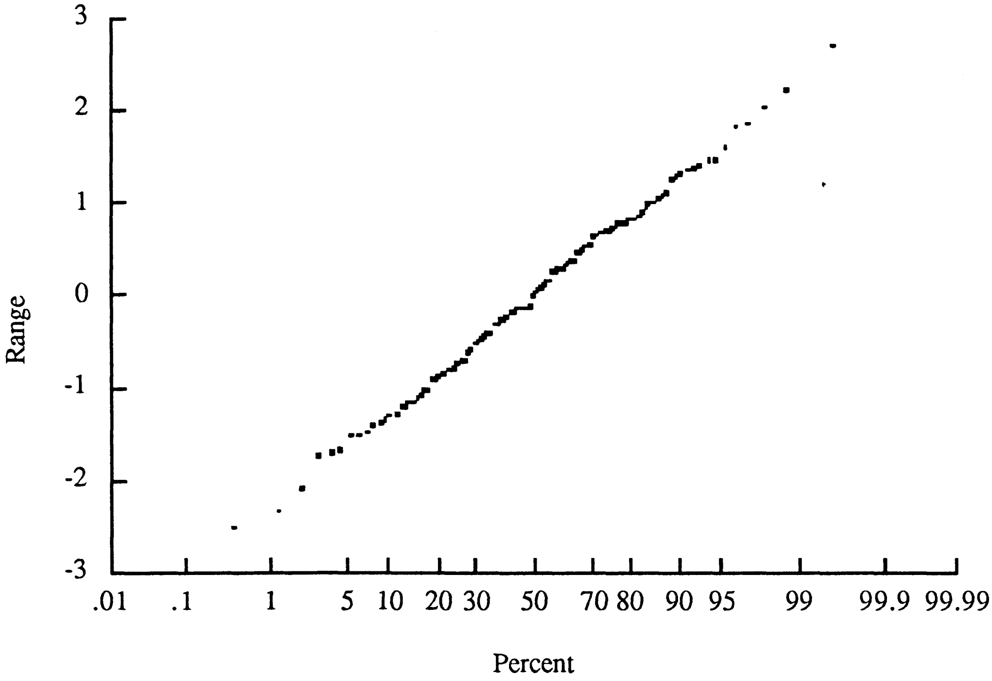


Figure 4.2.6 Normal Probability Plot of Log Transformation Model



4.3 Determining the Variance Structure

There are at least two major methodologies in the literature on variance function estimation: maximum likelihood estimation and least squares estimation. In the maximum likelihood approach, one assumes a parametric form of the variance function and finds the values of the parameters that maximize the function. One example of such approach can be found in Finney and Phillips [1977]. In the least squares approach, one does not make specific assumptions about the parametric form of the variance function. One simply performs regressions with the "response" being transformations of the residuals from a preliminary fit or sample standard deviations from replicates at a design point. (See Davidian and Carroll, 1987 for a more detailed discussion of this approach.)

As pointed out by Davidian and Carroll [1987], robustness plays an important role in the efficiency of variance function estimation. As the true distribution of the errors deviates from normal distribution, maximum likelihood estimation gradually loses its efficiency. Since the errors in our regression model may not be normally distributed, we use regression methods to estimate the variance function.

Many authors have proposed various approaches for variance function estimation through regression methods. The most popular approaches include: least squares on squared residuals, least squares on absolute residuals, and least squares on logarithm of absolute residuals.

The recommended procedure by Davidian and Carroll [1987] for the least squares approach is to find the parameters of the variance function iteratively. The generalized least squares method is used to estimate the parameters in the regression model, then the residuals are used to estimate the parameters of the variance function. The parameters of the variance function are used as the revised weights in the next round of generalized least squares estimation. This process is repeated until the parameters stabilize.

Since the focal point of this research is to develop an efficient algorithm for locating ATR sites optimally under non-constant variance models rather than identifying the exact form of the variance function, we omitted the iterative part of this approach. We used least squares on absolute residuals because it is the easiest to implement and appears to receive the most support in the literature.

We first fit the original regression model as described in Section 4.2. The absolute residuals of the regression model were then used as the response variable and the variables in the regression model were used as predictors. After eliminating insignificant variables, we found that absolute residual is proportional to the product of county population size and total number of through lanes on the road section. The regression results are summarized in Table 4.3.1. This variance function is used in Chapters 5 and 6 in conjunction with the development of numerical algorithms.

Table 4.3.1 Variance Function Using Absolute Residuals With Centered and Standardized Data

```
The regression equation is
ABSRES = 0.248 + 0.175 POP*LANE

Predictor        Coef       Stdev    t-ratio       p
Constant      0.24773     0.01826      13.57   0.000
POP*LANE      0.17459     0.01833       9.52   0.000

s = 0.2016      R-sq = 43.1%     R-sq(adj) = 42.6%

Analysis of Variance

SOURCE        DF          SS          MS         F        p
Regression     1      3.6883      3.6883     90.71    0.000
Error        120      4.8791      0.0407
Total        121      8.5673
```

4.4 Spatial Correlation and Possible Solutions

Most research in model-based sampling assumes that the population units generated from the superpopulation are independent and identically distributed. To some degree, this assumption is not met in this research. The traffic volume on one section of a road is correlated with those of its neighboring sections. In fact, it is possible that traffic volumes of two neighboring sections are perfectly correlated (if there is no intersection or interchange between them). Obviously, when traffic volumes on two sections are very highly correlated, it would be inefficient to install ATRs on both sections. As another illustration, suppose that a segment of interstate highway is closed for repairs for six months. The AADTs for all population units in this segment will be below their expected values. In this sense a positive correlation among these units is evident. We propose two solutions to this problem in the following two sections.

4.4.1 Methods Involving the Variance-covariance Matrix

One approach to dealing with the problem of spatial correlation is to explicitly specify the variance-covariance structure of traffic volumes for each pair of road sections. Thus, for example, if two sections are perfectly correlated, the variance-covariance matrix will force the information matrix to be singular and make the simultaneous selection of these two sections impossible. This approach, however, is very difficult to implement in the traffic-volume estimation problem. The variance-covariance matrix is usually very large and the covariances are not easily specified, particularly since they may depend on the time of day, day of the week, and so on.

Alternatively, one might model the pairwise correlation r_{ij} between sites i and j as a function of the distance d_{ij} between two sites. For instance, one might take

$$r_{ij} = e^{\lambda_1 d_{ij}}$$

where d_{ij} is the Euclidean distance between the sites. Under this method, we still have to compute the pairwise distance between two sites. Furthermore, two sites that are very far apart could have highly-correlated traffic-volumes.

No matter how we specify the variance-covariance matrix, these methods will increase the difficulty (and inversely affect the performance) of numerical algorithms since they require inversion of the n x n variance-covariance matrix. This procedure can be time consuming if the number of population units selected is large. We therefore need to investigate other methods that do not require this step.

4.4.2 Methods That Do Not Require the Variance-covariance Matrix

To avoid direct use of the variance-covariance matrix, we can simply prohibit pairs that are potentially highly correlated from being selected for installation of ATRs. There are several ways to implement this:

(1) If the (x,y) coordinate information for every data point is available, we can use this information to calculate the pairwise distances of population units. A critical value γ for the distance can be specified so that once a point is selected, all the points within radius γ of that point are excluded from consideration. Before attempting to bring a data point into the sample set, the algorithm would calculate pairwise distances between it and all the existing population units already in the sampled set. A candidate point will be added to the sample set only if all pairwise distances are greater than the critical value γ.

(2) If the (*x*,*y*) coordinate information is not available but prior experience and/or subjective judgment indicates that some population units should not be selected simultaneously, this information could be directly incorporated in the obvious way.

(3) The numerical algorithm can be designed to search only a subset of the population containing uncorrelated (or less correlated) population units. This subset could be provided by traffic engineers using their professional judgment and experience. An added advantage of this approach is that undesirable locations for ATRs can be excluded from consideration explicitly.

(4) In addition to the pre-selection method in (2), a post-selection method is also possible. Under this method, a solution is generated by the algorithm without any restrictions. A traffic engineer then examines the solution and makes substitution of population units if necessary. This is the approach that we recommend.

CHAPTER 5
CONSTRUCTING OPTIMAL SAMPLES

5.1 Introduction and Summary

In this research, our objective is to select *n* ATR sites from *N* road sections under certain optimality criteria. The problem arises because in most cases, *N* is much larger than *n*. (In our example, $N = 108{,}329$ and $n = 151$.) An exhaustive search for the optimal solution under a given criterion is theoretically possible, but very difficult and costly. Efficient algorithms are needed to construct optimal samples. We present two such algorithms in this research: an exchange algorithm, described in this chapter and a two-stage sampling algorithm, described in the next chapter.

In Section 5.2, we review previous work in optimal experimental design relevant to this report. Most research in model-based sampling has concentrated on the problem of estimating population totals. Usually the design criterion is to choose a strategy that minimizes the average mean squared error for predicting the population total. However, because AADTs are point-specific measures, the precision with which each individual AADT is predicted is the over-riding concern. One obvious optimality criterion that could be applied in this case is minimax: minimizing the maximum variance of prediction over all units in the population. But, as we show in Section 5.2, the minimax criterion is very difficult to use in the AADT estimation problem. An alternative criterion is proposed in Section 5.3. In Section 5.4, we review the literature of exchange algorithms. The development of an exchange algorithm based on the revised optimality criterion is presented in Section 5.5 followed by its evaluation in Section 5.6. In Section 5.7, we explore the possibility of using resampling techniques. Finally in Section 5.8, we discuss a faster version of the exchange algorithm for populations with special structures.

5.2 Review of Optimal Experimental Design and some Optimality Criteria

An N-point optimal design of experiments concerns the problem of taking observations at points x_i, $i = 1,\ldots, N$, from a compact design space χ to optimize a specified optimal criterion. An optimal design can be categorized as either exact or approximate. Roughly speaking, an exact design is when the numbers of observations taken at support points are all integers while an approximate design relaxes such a requirement.

Regression model f. Throughout, we shall assume we have the following linear model

$$\begin{aligned} &E(y_i) = f(x_i)^T \beta \\ &Var(y_i) = \sigma^2(x_i) = \sigma_i^2 \\ &Cov(y_i, y_j) = 0 \quad (i, j = 1, \ldots, N; i \neq j). \end{aligned} \tag{5.2.1}$$

The vector y_i is an N x 1 vector of observations, $f(x_i)^T$ is a p x 1 vector which specifies the assumed form of response, β is a p x 1 vector of unknown parameters, and σ_i^2 are assumed known.

Design Matrices: X is an N x p matrix with the i-th row containing $f(x_i)^T$. When rows corresponds to nonsampled cases are deleted, we obtain the sample design matrix X_s.

The Information Matrix. The matrix $X_s' V^{-1} X_s$ is called the information matrix, where X_s is an n x p matrix and V is the n x n variance-covariance matrix with diagonal entries of σ_i^2 and zero elsewhere. The normalized information matrix of a design ξ is denoted by

$$M(\xi) = \int_{\chi} f(x) f(x)^T \sigma^{-2}(x) d\xi(x).$$

The dispersion matrix, $M^{-1}(\xi)$, is sometimes written $D(\xi)$.

5.2.1 Measures of Optimality

D-optimality. This criterion was first suggested by Wald [1934] and given its name by Kiefer [1959]. Under this optimality criterion, a design is selected to maximize $|M(\xi)|$, the determinant of the information matrix.

G-optimality. Smith [1918] first considered designs that minimize the maximum variance of fitted values over the design space. A design ξ_G is G-optimal under constant variance errors if

$$\min_{\xi \in \Xi_\chi} \max_{x \in \chi} d(x, \xi) = \max_{x \in \chi} d(x, \xi_G)$$

where $d(x, \xi) = f(x)^T M^{-1}(\xi) f(x)$ is called the variance function.

A-optimality. Under this criterion, a design is selected to minimize the trace of the dispersion matrix. It is given the name of A-optimality because a design so selected also minimizes the average variance of the estimated regression coefficients.

I-optimality. Designs that minimize the integrated variance of fitted values over the design space are termed I-optimal. That is, I-optimal designs minimize

$$\int_{\chi} d(x, \xi) dx.$$

Cook and Nachtsheim [1982] discussed a similar optimality measure where designs are called I_λ-optimal if they minimize

$$\int_{\mathcal{X}} d(x,\xi)d\lambda(x).$$

The weighting function, λ, is specified by the experimenter. They also suggested that this optimality criterion is of special interest when the major concern of the experimenter is to make future predictions.

5.2.2 Optimality Criteria Related to This Research

Although the minimax (G-optimal) approach appears to be an appropriate optimality criterion in this research because it guards against the worst scenario, it is difficult to use computationally. Welch [1984] established some upper bounds for the variance function so the point with new maximum variance can be identified without searching through the entire design region. Even with this enhancement, Welch concluded that "...G optimality remains expensive. The G criterion is particularly prone to becoming trapped at a (poor) local optimum." (pp. 219-220)

We know of only two papers in the literature that deal with the problem of constructing sampling plans for the purpose of predicting responses associated with non-sampled units. Wynn [1977a] developed methods for constructing samples to minimize the maximum "normalized variance" of prediction, where the normalized variance of prediction at the point x_i is given by $var(\hat{y}_i) / \sigma_i^2$. He suggested an exchange algorithm as follows:

1. Form a sample of size n for which $X_s^T V^{-1} X_s$ is nonsingular, call it s_n.
2. Find $\max_{i \in s'} var(\hat{y}_i) / \sigma_i^2$.
3. Form a sample size of $(n+1)$ by adjoining unit j to the original sample, giving s_{n+1}.
4. Select a subsample of size n from s_{n+1}, called s_n^+, to maximize $|X_s^T V^{-1} X_s|$ over all samples s of size n in s_{n+1}.
5. Repeat steps 2-4 until no further improvements in $|X_s^T V^{-1} X_s|$ occur.

Wynn's algorithm requires only $(N\text{-}n)+(n+1) = N+1$ quadratic evaluations per exchange and the (possibly locally) optimal design can be generated iteratively. This is due to the near equivalence of the determinant and the minimax criterion considered. Wynn [1977b] also considered the problem of constructing sampling plans for large samples under the minimax criterion. This paper relaxed the requirement that the number of observations taken at support points be integers.

The minimax criterion in this research, however, is different from Wynn's in that we are not normalizing the variance of prediction. As a result, the rough equivalence to the determinant criterion disappears. The direct implication is that we can no longer cheaply generate the minimax design with iterative algorithms. Evaluation of the objective function requires that the variance of prediction be computed for each element of the population. For the extremely large population under consideration in this work, this leads to what may be insurmountable algorithmic difficulties.

Alternatively, one might consider minimizing the average variance of prediction over the nonsampled set

$$V(s') = \frac{1}{N-n}\left[\sum_{x_i \in s'} \sigma_i^2 + \sum_{x_i \in s'} f(x_i)^T (X_s^T V^{-1} X_s)^{-1} f(x_i)\right],$$

which is more economically computed as

$$V(s') = \frac{1}{N-n}\sum_{x_i \in s'} \sigma_i^2 + \frac{1}{N-n} TR\left[(X_s^T V^{-1} X_s)^{-1} W_{s'}\right]$$

where

$$W_{s'} = \sum_{x_i \in s'} f(x_i) f(x_i)^T,$$

s is the sampled set, s' is the nonsampled set, X_S is the design matrix for population units in the sampled set, and V is the variance-covariance matrix of points in the sampled set. As will be shown, in contrast with the maximum variance of prediction, this objective function can be evaluated economically.

Another advantage of using this optimality criterion is that design algorithms can be developed in parallel to the ones by Fedorov [1972]. The difference between our criterion and Fedorov's lies in the design region over which the objective function is evaluated. While Fedorov's criterion minimizes the objective function over a fixed space, ours minimizes the objective function only over the nonsampled set. The latter space is not fixed, since it obviously depends on s, the sampling plan. Designs that minimize $V(s')$ above will be termed V_S-optimal, following a very similar design criterion discussed in Welch [1984].

5.3 Review of Exchange Algorithms

Much research in optimal design theory has demonstrated the utility of exchange algorithms in constructing efficient designs. Exchange algorithms can be roughly divided into three categories:

(1) Algorithms that perform simultaneous exchanges of points between sampled and nonsampled sets. The original Fedorov [1972] algorithm is in this category.

(2) Algorithms that exchange points in sequence rather than simultaneously. The basic algorithm is to sequentially add a point to then delete a point from the current design. There are several variations to this basic algorithm. Mitchell's [1974] DETMAX algorithm allows "excursions" of up to k points from the original design where k is a user-specified value. Johnson and Nachtsheim's [1983] "k-exchange"

algorithm selects k points $x_{(1)}, x_{(2)}, ..., x_{(k)}$ with the lowest variance of prediction for deletion at each iteration. Each iteration is then broken down into k steps, one for each of the k points chosen for deletion. At the i-th step, $x_{(i)}$ is deleted and the point x^* that maximize variance of prediction over the design space given $(N - 1)$ points in the design is added to the design. (See Cook and Nachtsheim [1980] for descriptions of this and other related algorithms.)

(3) Algorithms that allow the addition of up to m points simultaneously to the initial design. Evans [1979] provided an example of such an algorithm.

In Johnson and Nachtsheim [1983], the authors rejected the idea of multiple point augmentation proposed by Evans [1979]. Their conclusion was that multiple point augmentation is in practice more likely to suffer from convergence to local optima and sequential single point augmentation generates comparably efficient designs in a fraction of computer time needed for Evan's algorithm. Because of these results, our exchange algorithms will be based on sequential single-point exchanges.

The basic steps of any sequential single-point exchange algorithm are as follows:

(1) Start with an initial n-point design. This design can be arbitrary (e.g. a random starting design) or carefully selected (e.g., via the Galil-Kiefer [1980] starting design algorithm) as long as it is nonsingular.

(2) Augment the design by adding one point. Add the point that causes the most improvement in the objective function value.

(3) Return the design to an n-point design by removing one point. The point removed is the one that causes the least "penalty" to the objective function value.

(4) If the exchange in steps (2) and (3) yields an improvement, go to step (2). Otherwise stop.

Although none of the exchange algorithms guarantees convergence to the global optimum, they represent an attractive approach to finding designs that are nearly optimal. In what follows, we present the development of an exchange algorithm for our V_S-optimality criterion.

5.4 Rank-1 Exchange Algorithm

Propositions 5.4.1 and 5.4.2 below indicate how the augmentation and deletion steps can be economically accomplished. Proofs of these two propositions can be found in Appendix A.

Proposition 5.4.1:

Assume the n-point plan s is nonsingular so that $\left(X_s^T V^{-1} X_s\right)^{-1}$ exists. Let $s_{new} = s + x_k$ for some $x_k \in s'$. Then

$$V(s'_{new}) = \frac{1}{N-n-1}\left[(N-n)V(s') - \Delta^{+}(x_k)\right]$$

where

$$\Delta^{+}(x_k) = \sigma_k{}^2 + \frac{1}{\sigma_k^2 + v_{kk}}\left[\sum_{x_i \in s'} v_{ik}{}^2 + \sigma_k^2 v_{kk}\right], \qquad (5.4.1)$$

σ_i^2 is the error variance of y at x_i, and $v_{ij} = f^T(x_i)\left(X_s^T V^{-1} X_s\right)^{-1} f(x_j)$.

Proposition 5.4.2:

Assume the n-point plan s is nonsingular. Let $s_{new} = s - x_h$ for some $x_h \in s$. Assuming s_{new} is nonsingular,

$$V(s'_{new}) = \frac{1}{N-n+1}\left[(N-n)V(s') + \Delta^{-}(x_h)\right]$$

where

$$\Delta^{-}(x_h) = \sigma_h{}^2 + \frac{1}{\sigma_h^2 - v_{hh}}\left[v_{hh}\sigma_h^2 + \sum_{i \in s'} v_{ih}{}^2\right], \tag{5.4.2}$$

$\sigma_h{}^2$ is the error variance at x_h, and $v_{ij} = f^T(x_i)\left(X_s^T V^{-1} X_s\right)^{-1} f(x_j)$.

Propositions 5.4.1 and 5.4.2 lead to the following exchange algorithm:

Algorithm 5.1: Rank-1 Exchange Algorithm

1. Form a sample s of size n for which $X_s^T V^{-1} X_s$ is nonsingular.
2. Add x_k to the sampled set where $x_k = \arg\max_{x_i \in s'} \Delta^{+}(x_i)$.
3. Update $X_s^T V^{-1} X_s$ and v_{ij}'s.
4. Delete x_h from the sampled set where $x_h = \arg\min_{x_i \in s} \Delta^{-}(x_i)$.
5. Update $X_s^T V^{-1} X_s$ and v_{ij}'s again.
6. Repeat steps 2-5 until no further improvements can be made to $V(s')$.

We shall call this algorithm the "Rank-1" exchange algorithm since it is accomplished through a series of rank-1 matrix updates. The major computational challenge of the algorithm comes from the calculation of $\sum_{x_i \in s'} v_{ij}^2$. The following proposition provides a very efficient numerical approach in computing this quantity.

Proposition 5.4.3

$$\sum_{x_i \in s'} v_{ik}^2 = TR\left\{A_k\left[\sum_{x_i \in pop} f(x_i) f(x_i)^T - \sum_{x_i \in s} f(x_i) f(x_i)^T\right]\right\}.$$

where

$A_k = Df(x_k)f(x_k)^T D$, a matrix depends only on the point being moved and D is the dispersion matrix, $\left(X_s^T V^{-1} X_s\right)^{-1}$.

Proof.

$$\sum_{x_i \in s'} v_{ik}{}^2 = \sum_{x_i \in s'} \left[f(x_i)^T Df(x_k) \right]^2 = \sum_{x_i \in s'} f(x_i)^T Df(x_k) f(x_k)^T Df(x_i)$$

Let $A_k = Df(x_k)f(x_k)^T D$, then

$$\begin{aligned} \sum_{x_i \in s'} v_{ik}{}^2 &= \sum_{x_i \in s'} f(x_i)^T A_k f(x_i) \\ &= TR\left\{ A_k \left[\sum_{x_i \in s'} f(x_i) f(x_i)^T \right] \right\} \\ &= TR\left\{ A_k \left[\sum_{x_i \in pop} f(x_i) f(x_i)^T - \sum_{x_i \in s} f(x_i) f(x_i)^T \right] \right\} \end{aligned}$$

The implications of this simple result for computing are clear. We need only compute $\sum_{x_i \in pop} f(x_i) f(x_i)^T$ (a p by p matrix) once and store it. Since $N >> n$, at each iteration, we only need to compute $\sum_{x_i \in s'} f(x_i) f(x_i)^T$ for population units in s. Thus, $\sum_{x_i \in s'} v_{ik}^2$ can be quickly updated by taking the product of two (p by p) matrices and computing the resultant trace.

5.5 Constructing Efficient Starting Designs

In order to increase the speed of the rank-1 exchange algorithm, we used a variant of the Galil-Kiefer [1980] starting algorithm for D-optimal designs to generate the starting design. This algorithm can be summarized as follows:

(1) The first point x_1 is selected either at random or to maximize the quantity $f(x_1)'f(x_1)$.

(2) For x_2 through x_p, we sequentially add population units that maximize $|XX^T|$ where X is the design matrix, (as suggested by Galil and Kiefer [1980].) Let $X_{i+1} = \begin{pmatrix} X_i \\ x_i^T \end{pmatrix}$.

It can be shown that

$$|X_{i+1}X_{i+1}^T| = |X_iX_i^T|\left[f(x_i)^T[I - X_i^T(X_iX_i^T)^{-1}X_i]f(x_i)\right] = |X_iX_i^T|q(x_i)$$

After the point that maximizes the quadratic form $q(x_i)$ has been selected, augment the matrix X and repeat step (2) until all p points are selected.

(3) If $q(x_i) > 0$ for $i = 2, ..., p$, X^TX will be nonsingular. Points $x_{p+1}, ..x_n$ are selected to maximize (5.4.1).

Experience with the above procedure suggests that the starting designs generated are often very close to the final optimal design. Usually the starting and final designs are different by only a few points and the average variance of prediction of two designs are very close to each other.

5.6 Empirical Evaluation

Since the rank-1 exchange algorithm is a heuristic and does not guarantee convergence to a global optimum, in what follows we study its performance in a small representative population where the globally optimal sample can be determined by exhaustive search. We use the following steps to study this problem:

(1) The data set of 122 ATR stations is used as a superpopulation.

(2) $N = 20$ units are randomly selected from this superpopulation as the test population and we use $n = 8$. For this small test population, there are $C_8^{20} = 125{,}970$ possible designs. A computer code was written to determine the optimal sample.

(3) The rank-1 exchange algorithm was used to generate a nearly optimal sample and the designs from steps (2) and (3) are compared.

Steps (2) and (3) were repeated 5 times to get some indication as to the consistency of the algorithm. The program used for this study is written in ANSI standard FORTRAN and run on a Cray X-MP/416 system. The sampling efficiency of the rank-1 exchange algorithm is computed as

$$E_v = \frac{V(s)_e}{V(s)_r}$$

where $V(s)_e$ is the minimum variance of prediction from exhaustive search and $V(s)_r$ is the average variance of prediction of the design generated by the rank-1 exchange algorithm. Results are summarized in the following table.

Run Number	# of Nonsingular Designs	$V(s)_e$	CPU time (Seconds)	$V(s)_r$ from Rank-1	CPU time for Rank-1 (Seconds)	Sampling Efficiency of Rank-1
1	58282	0.1083	152.81	0.1214	0.03	.892
2	30946	0.0810	134.59	0.0813	0.03	.996
3	67466	0.1161	148.29	0.1163	0.03	.998
4	51184	0.0712	142.02	0.0741	0.03	.961
5	26312	0.0956	127.19	0.0971	0.03	.985

With the exception of the first test population, the rank-1 exchange algorithm consistently generated designs that exceed 96% sampling efficiency even though none of the final designs generated are globally optimal. The major advantage of the rank-1 exchange algorithm over the exhaustive search, however, is the speed with which it constructs nearly optimal designs. In each case, it took the rank-1 exchange algorithm only 0.03 seconds to generate a final solution.

From the numerical studies above, the rank-1 exchange algorithm appears to generate efficient designs. Its speed, however, is a source of concern for large *N*. This is because at each iteration, (*N* - *n*) evaluations of the quantity (5.4.1) and *n* evaluations of the quantity (5.4.2) are needed. In next two sections, we discuss possible modifications of the rank-1 exchange algorithm to reduce required computation. In Section 5.7, we discuss the possible combination of resampling techniques and the rank-1 exchange algorithm. In the same section, we also discuss the use of repeated simple random sampling as yet another "resampling" approach. In Section 5.8, we discuss a much faster version of the rank-1 exchange algorithm when the population of interest has a particular structure.

5.7 Application of Resampling Techniques

Since the time required by the rank-1 exchange algorithm can be expected to increase with *N*, one intuitive approach is to use a subpopulation for optimal sample selection. The basic steps of this approach can be described as follows:

Algorithm 5.2: Rank-1 Exchange Algorithm applied to Subpopulations

(1) First, a subpopulation of size N_S is selected from the population of size *N*.

(2) The rank-1 exchange algorithm is applied to the subpopulation, taking advantage of its smaller size. An optimal sample is generated with respect to the subpopulation.

(3) Repeat steps (1) - (2) *r* times and use the generated sample having the best objective function value as the final solution.

Another "resampling" approach based on repeated simple random sampling is described below.

Algorithm 5.3: Repeated Simple Random Sampling

(1) A simple random sample of size n is selected from the population of size N and the average variance of prediction is calculated for this sample.

(2) Step (1) is repeated r times. The design having the smallest average prediction variance is chosen to be the optimal sample.

The effectiveness and applicability of both approaches will be investigated in Chapter 7 when we compare the performance of various algorithms.

5.8 Special Rank-1 Exchange Algorithm for Populations with Special Structure

The efficient implementation of the rank-1 exchange algorithm discussed in Sections 5.4 and 5.5 makes it suitable for medium to large size problems. When the population of interest consists of population units that can be classified entirely by categorical variables, further improvement of the rank-1 exchange algorithm is possible. Let l_i denote the number of levels of the i-th categorical variable. Suppose there are q categorical variables, there are at most $N_d = \prod_{i=1}^{q} l_i$ distinct population units x. Let N_i denote the number of identical x vectors and n_i the number of points sampled of the i-th type, x_i, for i = 1, ..., N_d. The rank-1 exchange algorithm can proceed with the following changes:

(1) N_i, n_i, and x_i are determined for $i = 1,\dots, N_d$.

(2) The average variance of prediction can be computed as

$$V(s') = \frac{1}{N-n}\left[\sum_{i=1}^{N_d}(N_i - n_i)\sigma_i^2 + \sum_{i=1}^{N_d}(N_i - n_i)f(x_i)^T M(s)^{-1} f(x_i)\right]$$

where

$$M(s) = \sum_{i=1}^{N_d} n_i f(x_i) f(x_i)^T v^{-1}(x_i).$$

(3) Augmentation both for the rank-1 exchange and the Galil-Kiefer starting design algorithm result from a search over the set of N_d distinct vectors rather than the set of N population units. This will obviously result in a reduction of search time by a factor of $\frac{N_d}{N}$.

The success of this modification rests on the premise that levels are categorical and N_d *is* small compared to N. This happens to be true in this research. Recall our regression model contains four distinct predictors: county population size, number of traffic lanes, ownership of the road (state/nonstate), and rural/urban code. County population size and number of traffic lanes were treated as continuous variables in Chapter 4. However, after we randomly selected 5,000 road sections from the true population and plotted these two "continuous" variables against each other under each combination of state/nonstate and rural/urban codes (Figures 5.1 through 5.4), it is clear that these two variables can also be treated as categorical variables. For instance, in Minnesota there are 87 values for county population size and 10 possible values for the number of traffic lanes, then each unit in the population must belong to one and only one of the 3,480 (87 x 10 x 2 x 2) types. At each step of the starting and exchange processes, the computer code checks only the 3,480 distinct x's instead of the 108,329 original population units, this improvement therefore should result in approximately 96% saving in search time. The only additional step in this faster algorithm is (1), in which population units are classified by type.

As noted, these savings are possible only if the points in a population can be classified entirely by categorical variables. When there are truly continuous variables (or when there are too many categories associated with these variables), we have to rely on other techniques such as clustering to reduce the dimensionality of the problem. We discuss the application of such techniques in connection with approximate designs in the next chapter.

Figure 5.1 County Population Size vs. Number of Traffic Lanes for Rural Nonstate Roads

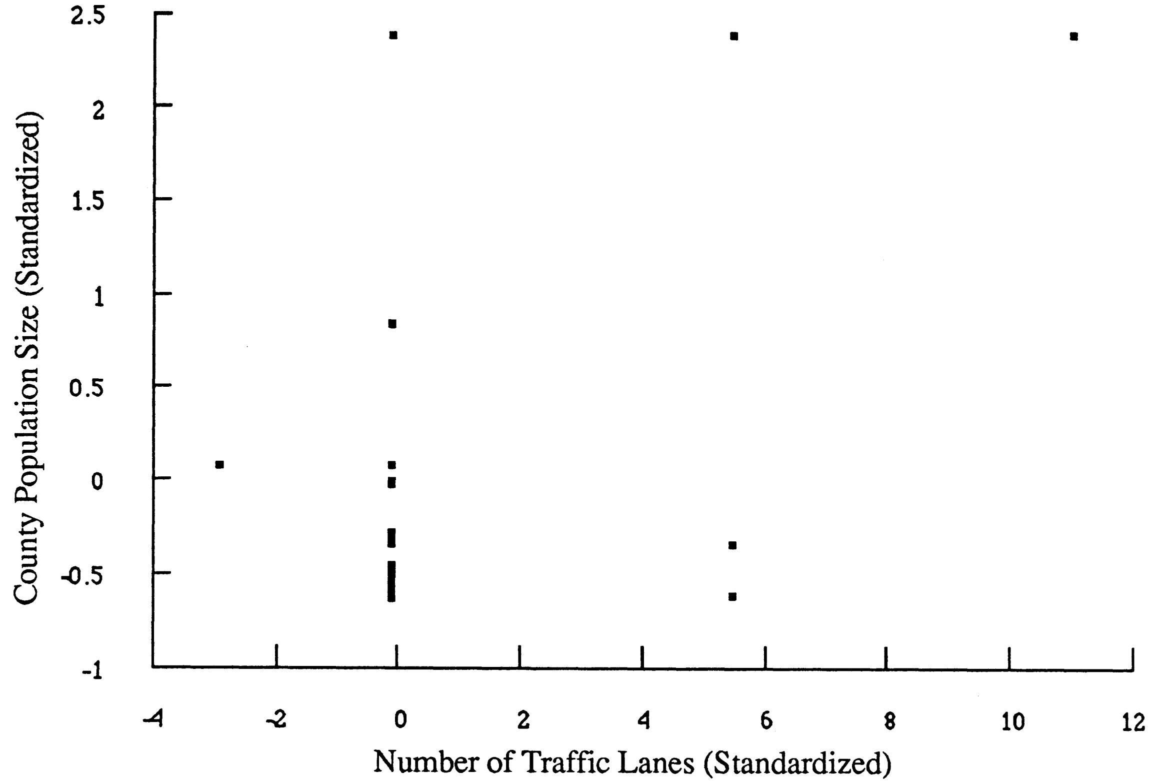


Figure 5.2 County Population Size vs. Number of Traffic Lanes for Rural State Roads

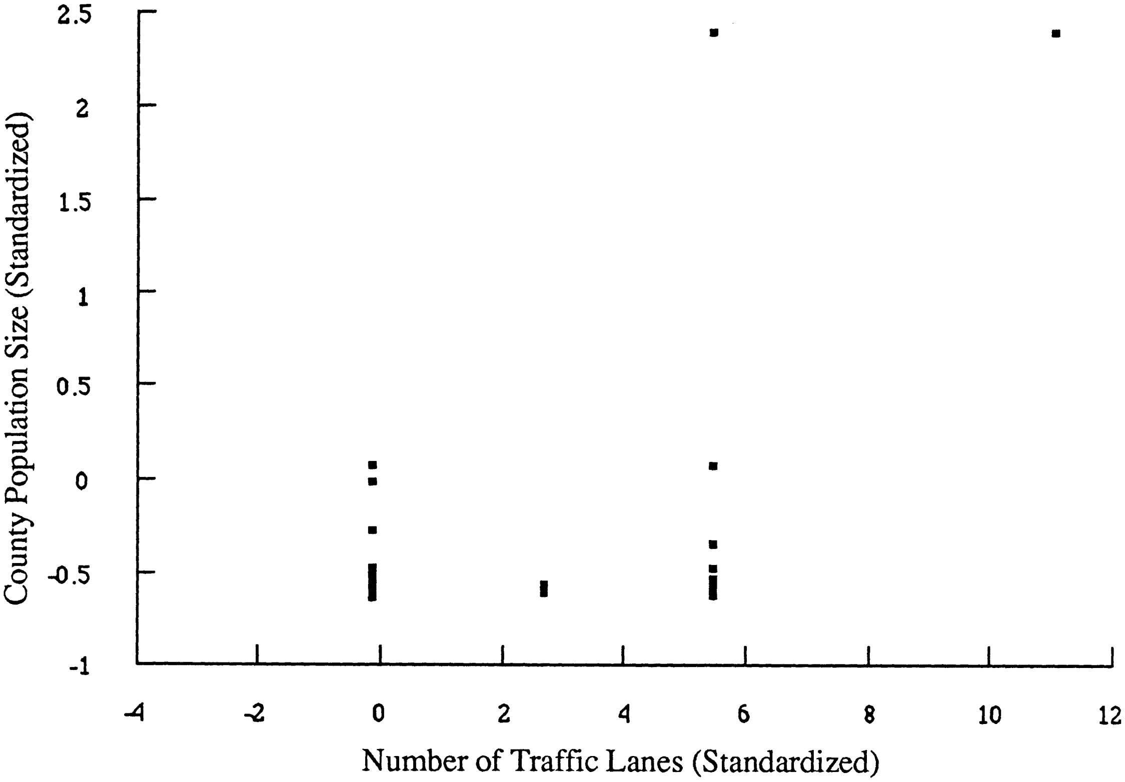


Figure 5.3 County Population Size vs. Number of Traffic Lanes for Urban Nonstate Roads

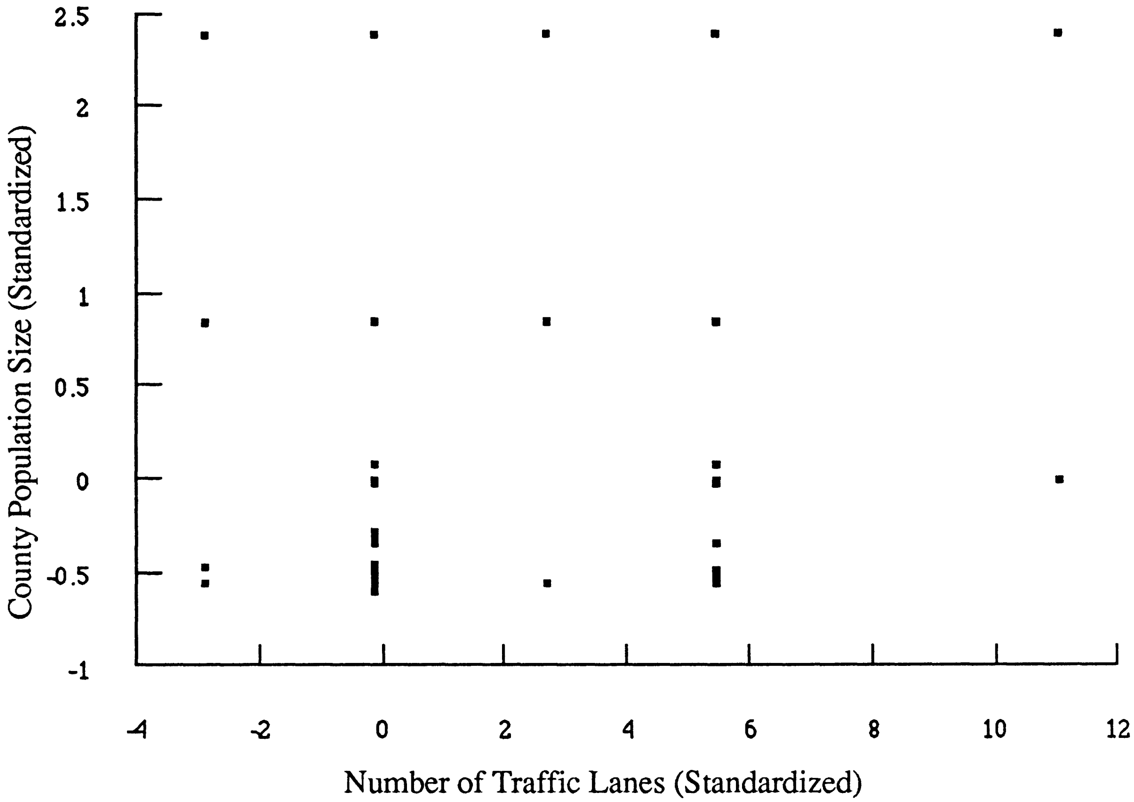
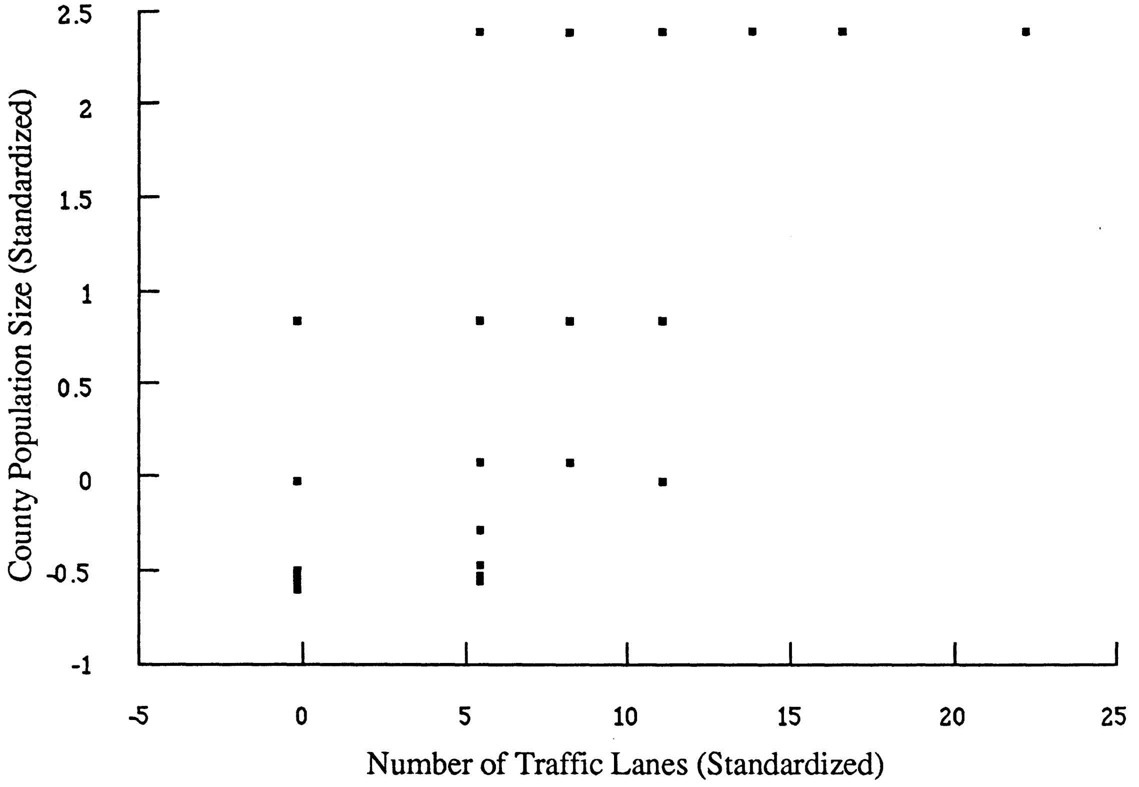


Figure 5.4 County Population Size vs. Number of Traffic Lanes for Urban State Roads



CHAPTER 6
CONSTRUCTING NEARLY OPTIMAL SAMPLES

6.1 Overview of the Two-stage Approach

In Chapter 5, we developed a single-point exchange algorithm for constructing optimal samples for small to moderately sized populations. We also discussed a simple modification for populations with categorical units. In this chapter, we develop an alternative two-stage sampling algorithm for use with large populations. This two-stage algorithm combines clustering and large-sample approximate design techniques. During the first stage, the population units are grouped into k clusters based on the similarity of data attributes. Cluster centroids are then used to represent the original points. In the second stage, large sample approximate design techniques are used to calculate the optimal weight for each cluster. These optimal weights indicate approximately how many points should be selected from each cluster. The approximate design is then rounded off to an exact design and points are selected from within each cluster using either simple random sampling or optimal sampling.

One advantage of this two-stage approach is that its performance is less affected by the problem size than the original rank-1 exchange algorithm. This is because the computer time needed for the second stage of this algorithm is a factor of k (number of clusters) rather than N (population size.) If clustering can be achieved economically, this approach can be expected to hold a significant speed advantage over the rank-1 exchange algorithm.

On the other hand, since we represent a group of points by its centroid, the effectiveness of this two-stage approach depends on the homogeneity of the points within each cluster. Sampling efficiency of the two-stage approach therefore becomes an important issue. We will study this issue, along with others, in Chapter 7 when we evaluate various algorithms.

In Section 6.2, we develop a large-sample "approximate" design criterion. Designs that are optimal by this criterion are developed analytically in some cases. An numerical algorithm is also developed. In Section 6.3, we describe the implementation of the cluster analysis in this project. Finally in Section 6.4, we study the performance of the two-stage sampling algorithm.

6.2 Large Sample Approximate Design

A large sample approximate design using the minimax criterion was discussed in Wynn [1977b]. In this paper, Wynn generalized the exact theory of minimax design he developed earlier to include the continuous case. As noted previously, there are two major differences between our approach and Wynn's approach:

(1) Since our regression analysis in Chapter 4 has indicated nonconstant error variances, we relax the constant variance assumption used by Wynn.

(2) We use a different optimality criterion in this research. Because of the computational difficulties associated with the minimax criterion (see Chapter 5), we focus on V_S-optimality.

Suppose the population has been partitioned into k clusters. Let χ denotes the sample space. That is,

$$\chi = \bigcup_{i=1}^{k} \chi_i$$

where χ_i represents the set of population units in the i-th cluster.

$$\chi_i = \{x_{ij}, j=1,\dots,N_i\},$$

N_i is the number of population units in cluster i, $i=1,\dots, k$. Next, denote the centroid of the i-th cluster by:

$$\bar{x}_i = \sum_{j=1}^{N_i} \frac{x_{ij}}{N_i}$$

Let n_i denote the number of observations to be taken from the i-th cluster so that $n = \sum_{i=1}^{k} n_i$. Assuming N is large ($N >> n$), we need to determine the proportion of observations in the population, $\xi(\bar{x}_i) = \frac{n_i}{N}$, that will come from the i-th cluster.

Let $\xi_0(\bar{x}_i) = \frac{N_i}{N}$ denote the proportion of population units in cluster i. Note that,

$$\sum_{i=1}^{k} \xi(\bar{x}_i) = F = \frac{n}{N} < 1,$$

where F is the sampling fraction. Thus ξ is not a probability measure and therefore not a standard design measure as used in the optimal design literature. Following Wynn [1977b], let

$$\underline{B}(\xi) = \left\{\bar{x}_i \middle| \xi_0(\bar{x}_i) > \xi(\bar{x}_i) \geq 0\right\}$$

denote the set of cluster centroids from clusters having unsampled population units and let

$$\Xi(\xi_o, F) = \left\{\xi \middle| 0 \leq \xi(\bar{x}_i) \leq \xi_o(\bar{x}_i), \textstyle\sum_{i=1}^{k} \xi(\bar{x}_i) = F\right\}$$

denote the space of all admissible designs. The sampling design problem is to find $\xi \in \Xi(\xi_o, F)$ that optimizes the criterion of interest.

In this project, we concentrate on the V_S-optimality criterion (discussed in Chapter 5). The analytical form of which is given in Section 6.2.1, followed by numerical construction of large sample V_S-optimal sampling plans.

6.2.1 Large Sample V_S-optimal Sampling Plans

In this section, we first derive an expression for the average variance of prediction in terms of a continuous design. Recall that in the exact design situation (Chapter 5), we have

$$V(s') = \frac{1}{N-n}\left[\sum_{x_i \in s'} \sigma_i^2 + \sum_{x_i \in s'} f(x_i)^T M^{-1}(s) f(x_i)\right].$$

In the approximate design situation, the average variance of prediction can be reexpressed as follows:

$$V(\xi) = \frac{1}{N-n}\left[N\left(E_{\xi_0}\sigma_{\bar{x}}^2 - E_\xi\left[\sigma_{\bar{x}}^2\right]\right) + TR\left[M^{-1}(\xi)W_{\xi_0}\right] - TR\left[M^{-1}(\xi)W_\xi\right]\right], \qquad (6.2.1)$$

where

$$W_\xi = \sum_{i=1}^{k} f(\bar{x}_i) f^T(\bar{x}_i)\xi(\bar{x}_i) = E_\xi\left[f(\bar{x}) f^T(\bar{x})\right],$$

and

$$W_{\xi_0} = E_{\xi_0}\left[f(\bar{x}) f^T(\bar{x})\right].$$

Analytical characterization of V_S-optimal sampling plan is difficult unless the model and sample spaces take very simple forms. We present the numerical construction of V_S-optimal sampling plan instead.

6.2.2 Numerical Construction of Large Sample V$_S$-optimal Sampling Plans

Using the notation in Section 6.2.1,

$$V(\xi) = \frac{1}{N-n}\left[N\left(E_{\xi_0}[\sigma_{\bar{x}}^2] - E_{\xi}[\sigma_{\bar{x}}^2]\right) + TR\left[M^{-1}(\xi)W_{\xi_0}\right] - TR\left[M^{-1}(\xi)W_{\xi}\right]\right].$$

An approximate design algorithm can be developed as follows:

(1) At iteration j, we have a nonsingular design ξ_j.

(2) Generate a new design by shifting weight from clusters in the old design to the cluster having centroid $\bar{x}_j^*$ via $\xi_{j+1} = (1-\alpha_i)\xi_j + \alpha_i \xi_{\bar{x}_j^*}$. $\xi_{\bar{x}_j^*}$ is a point-mass measure equal to F if $x = \bar{x}_j^*$ and 0 elsewhere. If $\bar{x}_j^*$ is suitably chosen, improvements will result.

(3) Repeat step (2) until no further improvement is possible.

The major task in this algorithm is to determine $\bar{x}_j^*$(optimal cluster) and α_j (weight shifted) at each iteration. It can be shown (Appendix B) that:

<u>Proposition 6.2.1</u>

$\bar{x}_j^*$ should be chosen to maximize

$$\sigma_{\bar{x}^*}^{-2}\phi\left(\bar{x}^*, \xi_i, W_{\xi_0} - W_{\xi_j}\right) + N\sigma_{\bar{x}^*}^2 + d\left(\bar{x}^*, \xi_j\right),$$

where

$$\phi(x,\xi,W) = f^T(\bar{x})\left[M^{-1}(\xi)WM^{-1}(\xi)\right]f(\bar{x})$$

$$d(x,\xi) = f^T(\bar{x})M^{-1}(\xi)f(\bar{x})$$

and

$$W_{\xi} = \sum_{i=1}^{k} f(\bar{x})f(\bar{x})^T\xi(\bar{x}).$$

As to the determination of α_j, the sequence $\alpha_j = \frac{1}{k+j}$ works well in practice for approximate design algorithms although it does not give a monotone decreasing sequence of $V(\xi)$.

Algorithm 6.1: Approximate Design Algorithm for Determining Cluster Weights

From the above, we can develop an approximate design algorithm as follows:

(1) Assume a non-degenerate starting design, ξ_1, exists. Set $j = 1$, compute $M^{-1}(\xi_1)$, W_{ξ_1}.

(2) Set $\alpha_j = \frac{1}{k+j}$ for $k > 0$.

(3) Find $\underset{\bar{x}^* \in \underline{B}(\xi_i)}{Max} \gamma(\bar{x}_i) = \gamma(\bar{x}_j^*)$

(4) Form $\xi_{j+1} = (1-\alpha_j)\xi_j + \alpha_j \xi_{\bar{x}_j^*}$, update $M^{-1}(\xi_j)$, $V(\xi_j)$, $W_{\xi_{j+1}}, \gamma(\bar{x}_i)$ $i = 1,\ldots,k$.

(5) If $V(\xi_j)$ - $V(\xi_{j+1})$ is sufficiently small, stop. Otherwise, set $j = j+1$, go to step 3.

The optimal weights computed by the algorithm above determine the number of points to be selected from each cluster. In this project, we develop two algorithms for the two-stage sampling approach.

Algorithm 6.2: Clustering followed by Random Sampling Within Clusters

(1) Points in the population of interest are clustered.

(2) Optimal weight for each cluster is calculated by the approximate design algorithm.

(3) The required numbers of points are randomly selected from within clusters based on the optimal weights from step (2).

Algorithm 6.3: Clustering followed by Optimal Selection within Clusters

(1) Points in the population are clustered.

(2) Optimal weight for each cluster is calculated by the approximate design algorithm.

(3) Random samples are taken from each cluster.

(4) The rank-1 exchange algorithm (discussed in Chapter 5) is used within each cluster to further reduce the average prediction variance.

Clustering--which is the first step shared by both algorithms--will be reviewed in the next section. The performances of these two algorithms will be evaluated in Chapter 7.

6.3 Implementation of Cluster Analysis in This Project

In this section, we describe the implementation of cluster analysis in this project. A typical clustering process includes seven steps (Milligan and Cooper [1987]):

(1) Selection of entities to be clustered.

(2) Selection of variables used in clustering.

(3) Standardization of data if necessary.

(4) Selection of a similarity (e.g. correlation) or dissimilarity (e.g. distance) measure.

(5) Selection of a clustering method.

(6) Determination of the number of clusters.

(7) Interpretation of the resulting cluster analysis.

These seven steps are implemented in this project as follows:

(1) Entities Selected:

all road sections in the State of Minnesota.

(2) Variables used for clustering:

We use the same variables used in constructing the regression model (including interaction and second-order terms.) Quantitative variables include: county population size, number of traffic lanes (both directions), interaction term of county population size and number of traffic lanes as well as the squared term of the traffic lanes. Qualitative variables include rural-urban code and ownership of the section (state or non-state road.)

(3) Variable Standardization:

Because of the large differences in the scales of the quantitative variables (county population vs. number of traffic lanes), we standardize all quantitative variables.

(4) Similarity /Dissimilarity Measure:

Euclidean distance is used as the dissimilarity measure.

(5) Clustering Method:

In this research, a fast version of the K-means algorithm called the "Nearest Centroid sorting" is used to cluster the points. Since the resulting clusters depends to some extent on the selection of the initial seeds and the K-means methods are known to produce unsatisfactory results when the initial seeds are chosen poorly (Milligan [1980]; Milligan and Cooper [1987]), we used the FASTCLUS procedure in SAS User's Guide: Statistics to select initial seeds (pp. 378-379.)

(6) Number of Clusters:

There is an implicit lower bound on the number of clusters when we use the two-stage sampling proposed in this project. Since our regression model contains seven unknown parameters (including the intercept term), we must have at least 7 clusters in order to obtain a nonsingular design, even if the true number of clusters in the population is less than 7. In practice, we have found that use of the Calinski and Harabasz [1974] index does not necessarily lead to a most efficient sampling plan (see Section 6.4.1). We therefore recommend that the algorithm be run for a range of numbers of clusters.

6.4 Evaluation of the Two-stage Sampling Algorithm

In this section, we evaluate the performance of the two-stage sampling algorithm in two areas. First, we study how the choice of the number of clusters affect the performance of two-stage sampling. Second, we estimate the efficiency of two-stage sampling when compared to simple random sampling. To study these two problems, we used the data set corresponding to the 122 ATR stations. This is the same data set we used to construct the regression model in Chapter 4.

6.4.1 Relationship Between Average Variance of Prediction of the Optimal Sample and Number of Clusters

A simulation study was conducted to investigate how the choice of the number of clusters affects the selection of the optimal sample. In this study, $N = 122$, $n = 30$, and the number of clusters included in the study ranged from 7 to 28. For each specified number of clusters, the following steps were taken:

(1) The ATR data set of 122 road sections was clustered into the required number of clusters.

(2) Optimal weights for each cluster were computed using the approximate design algorithm (algorithm 6.1) described in Section 6.2. We then rounded off the approximate design and computed the number of road sections to be selected from each cluster.

(3) From within each cluster, a simple random sample was drawn. The average variance of prediction was computed for the resultant sample.

(4) Step (3) was repeated 5,000 times and the mean, maximum value, minimum value, and standard deviation of the 5,000 average prediction variances were recorded.

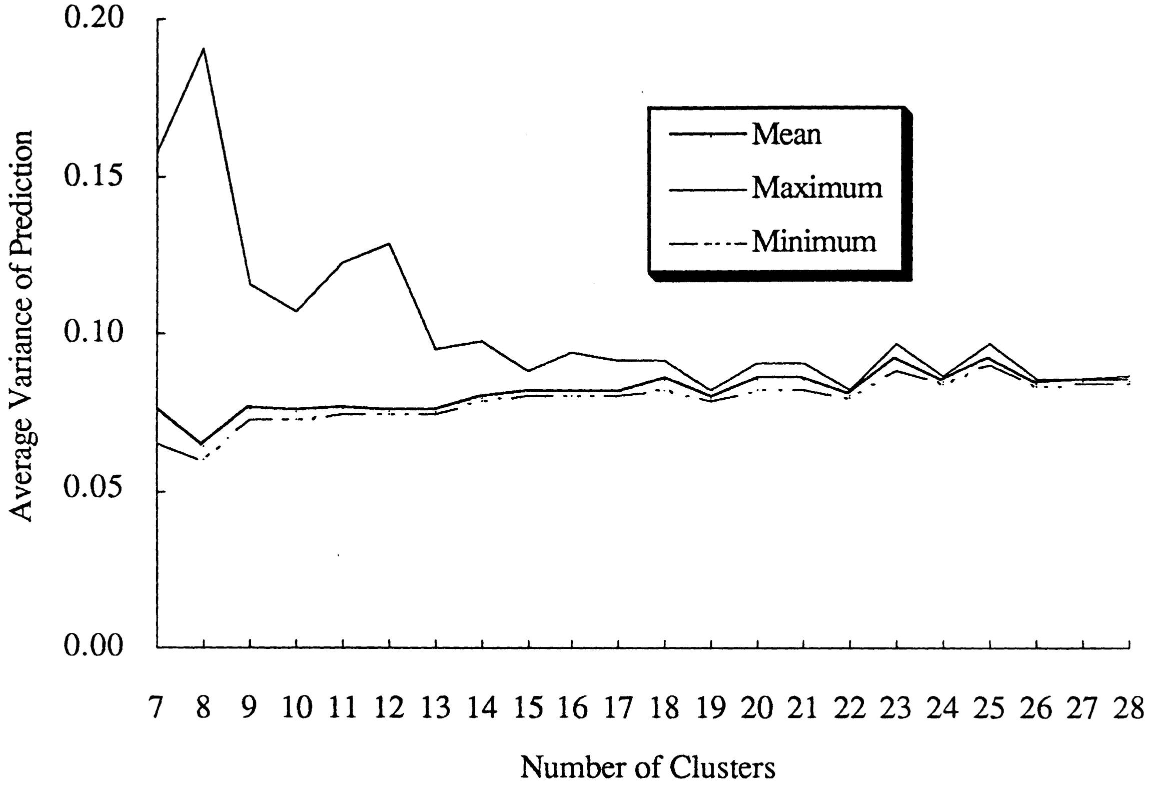
As we can see in Table 6.4.1 and Figure 6.4.1. The range of the average prediction variance becomes stable and very small after the number of clusters reached 18. This is because as the number of clusters increases, the number of points in each cluster decreases. As a result, the samples are quite similar. However, it should be noted that the range of the average prediction variance was not a monotone decreasing function of the number of clusters. For example, the range of the average prediction variance was the largest when the number of clusters specified was 8. We see another increase in the average prediction variance with 11 and 12 clusters used. These three instances represent the situation where clustering results in groups that are not homogeneous. In other words, there are "outliers" in some of these clusters, resulting in unusual values of the average prediction variances when the outliers are selected. In general, the average variance of prediction increases as the number of clusters increases. This is probably because of the stopping rule and the round-off procedure used in the two-stage sampling approach. As the number of clusters increases, the weight of each cluster decreases. Some "important" points may not be selected if each of them is in a cluster by itself.

It is also interesting to note that partitioning the population into 8 clusters generated both the smallest and largest average variance of prediction. This means in this particular example, if we cluster the ATR data set into 8 clusters and select a simple random sample, the sample can be either very good or very bad (compared to samples obtained using different number of clusters.) In this case, optimal selection within clusters (algorithm 6.3) may be helpful in reducing the average prediction variance by replacing the "bad" outliers in clusters.

Table 6.4.1 Relationship Between Average Variance of Prediction and Number of Clusters in Two-stage Sampling using Clustering Algorithm 1 - Update after Each Loop

Number of Clusters	Mean V(ξ)	Maximum V(ξ)	Minimum V(ξ)	Std. Deviation of V(ξ)
7	0.0763	0.1577	0.0649	0.0041
8	0.0647	0.1913	0.0599	0.0053
9	0.0763	0.1161	0.0732	0.0015
10	0.0757	0.1073	0.0726	0.0015
11	0.0764	0.1229	0.0745	0.0015
12	0.0759	0.1288	0.0745	0.0013
13	0.0758	0.0951	0.0747	0.0010
14	0.0800	0.0977	0.0786	0.0009
15	0.0821	0.0880	0.0810	0.0007
16	0.0821	0.0941	0.0807	0.0008
17	0.0821	0.0914	0.0809	0.0007
18	0.0858	0.0913	0.0826	0.0014
19	0.0800	0.0826	0.0785	0.0006
20	0.0857	0.0910	0.0824	0.0013
21	0.0858	0.0911	0.0824	0.0013
22	0.0808	0.0825	0.0796	0.0005
23	0.0921	0.0966	0.0886	0.0013
24	0.0850	0.0868	0.0839	0.0005
25	0.0922	0.0972	0.0902	0.0011
26	0.0843	0.0861	0.0830	0.0005
27	0.0848	0.0859	0.0839	0.0003
28	0.0851	0.0865	0.0842	0.0004

Figure 6.4.1 Relationship Between Number of Clusters and Average Variance of Prediction in Two-stage Sampling (5,000 trials for each number of clusters)



*ATR data set used; N = 122, n = 30.

6.4.2 Efficiency of Two-stage Sampling Algorithm

In this section, we estimate the relative sampling efficiency of simple random sampling vs. two-stage sampling. (See Section 5.6 for a definition of sampling efficiency.) The ATR data set was used for evaluating these two approaches. We used N = 122 and n = 30.

In simple random sampling, a sample of required size is chosen from the test population and the average variance of prediction is calculated. This procedure is performed repeatedly in order to assess the sampling distribution of the average prediction variance. We varied the number of simulated samples N_{sim} simply to assess its effect on the minimum and maximum average prediction variances found with SRS. In two-stage sampling, we again used algorithm 6.2 (random selection within clusters). We also varied N_{sim} to study its effect on the sampling distribution of $V(\xi)$.

Table 6.4.2 summarizes the results from partitioning the ATR data set into 10 clusters in two-stage sampling (10 was an arbitrarily chosen number.) Table 6.4.3 contains results of the efficiency study when we partition the same data set into 8 clusters. (We chose this particular number to study further because in section 6.4.1, both very good and very bad samples were generated by using 8 clusters.) Some observations follow from these two tables:

(1) On average, the two-stage approach has significantly smaller average variance of prediction than simple random sampling.

(2) As expected, the two-stage approach has much smaller *variation* in average variance of prediction. This means, of course, that we are unlikely to select a "bad" sample with this approach.

(3) The relative sampling efficiency of the simple random sampling approach for the ATR population is about 24%. Furthermore, simple random sampling generated singular designs about 5% of the time. These cases were ignored in the computation of sampling efficiency.

When compared to the design generated by the rank-1 exchange algorithm, designs generated using algorithm 6.2 have a relative sampling efficiency of anywhere between 58% to 68% in this particular study. A possible improvement is to use algorithm 6.3 (optimal selection within clusters) at the cost of increased computation. The performance of algorithm 6.3, as well as the performance of rank-1 exchange vs. two-stage sampling algorithms will be compared more thoroughly in terms of computer time requirements in Chapter 7.

Table 6.4.2 Comparison of $V(s')$ for the Rank-1 Exchange Algorithm, Two-stage Sampling Approach, and Simple Random Sampling (N = 122, n = 30)

Average Variance of Prediction of Various Approaches									
Number of Simulated Samples	Simple Random Sampling				Clustering* + Optimal Weighting/ SRS within cluster				Efficiency of SRS
	Max.	Min.	Avg.	S.D.	Max.	Min.	Avg.	S.D.	
20	3.190	.096	.391	.695	.078	.073	.076	.001	.1934
100	3.190	.096	.299	.473	.080	.073	.076	.001	.2523
500	3.564	.092	.322	.481	.083	.073	.076	.001	.2351
1000	8.189	.086	.331	.550	.085	.073	.076	.001	.2284
2000	8.189	.086	.322	.502	.087	.073	.076	.001	.2351
5000	8.189	.077	.314	.476	.107	.073	.076	.002	.2412
Optimal Sampling Via Rank-1 Exchange Algorithm: .044									

* The test population containing 122 ATR stations was clustered into 10 clusters.

Table 6.4.3 Comparison of $V(s')$ for the Rank-1 Exchange Algorithm, Two-stage Sampling Approach, and Simple Random Sampling (N = 122, n = 30)

Average Variance of Prediction of Various Approaches									
Number of Simulated Samples	Simple Random Sampling				Clustering* + Optimal Weighting/ SRS within cluster				Efficiency of SRS
	Max.	Min.	Avg.	S.D.	Max.	Min.	Avg.	S.D.	
20	3.190	.096	.391	.695	.077	.062	.064	.004	.1650
100	3.190	.096	.299	.473	.100	.061	.065	.005	.2171
500	3.564	.092	.322	.481	.117	.061	.065	.006	.2024
1000	8.189	.086	.331	.550	.118	.060	.065	.005	.1953
2000	8.189	.086	.322	.502	.121	.060	.065	.005	.2014
5000	8.189	.077	.314	.476	.191	.060	.065	.005	.2062
Optimal Sampling Via Rank-1 Exchange Algorithm: .044									

* The test population containing 122 ATR stations was clustered into 8 clusters.

CHAPTER 7
EVALUATION OF ALGORITHMS AND RECOMMENDATIONS FOR USE

7.1 Introduction and Conclusion

In this chapter, we compare the performance of the five numerical algorithms presented in Chapters 5 and 6. The comparisons will be based on the sampling efficiency and the execution time required for each approach. The sampling efficiency of an algorithm is defined as a ratio similar to the one presented in Section 5.6, except we use the average prediction variance of the rank-1 exchange algorithm as the benchmark. Because we use a test population containing 5,000 road sections, it is impossible to identify the globally optimal design through exhaustive search.

The ANSI standard FORTRAN program used for evaluation is presented in Appendix C. Computing was done on Minnesota Supercomputer Institute's Cray X-MP/416 system running UNICOS 5.1. This is a four-processor vector machine, with a 9.5 nanosecond clock period, capable of 850 megaflops at peak performance.

The five algorithms included in this study are:

(1) the rank-1 exchange algorithm (algorithm 5.1),

(2) resampling, based on repeated application of the rank-1 exchange algorithm (algorithm 5.2),

(3) resampling, based on repeated simple random sampling (algorithm 5.3),

(4) two-stage sampling with random selection within clusters (algorithm 6.2), and

(5) two-stage sampling with optimal selection within clusters (algorithm 6.3).

Results indicated that the rank-1 exchange algorithm generates the most efficient designs while two-stage sampling with random selection within clusters is the fastest algorithm that generates efficient designs. From our test results in this chapter, it is concluded that the rank-1 exchange algorithm should be used if a highly efficient design is required or when the population of interest contains categorical population units. On the other hand, if the population of interest has no special structure and a good design needs to be generated quickly, then two-stage sampling is an attractive alternative.

Numerical results for procedures (1)-(5) are reported in Sections 7.3-7.7, respectively. Discussions and recommendations are given in Section 7.8. The test population used is described in the next section.

7.2 Description of the Test Population

The test population consists of 5,000 road sections randomly selected from the population of 108,329 Minnesota road sections. We used the test population instead of the entire population to conserve computing resources. Comparisons of the characteristics of the two populations are provided in Table 7.2.1. From this table, it can be seen that the test population is highly representative of the true population. The first two moments of all four variables for both populations are quite close. The only notable difference between the two populations is that none of the sections with seven or more traffic lanes was selected from the true population to be included in the test population.

One critical issue of using the test population is the extrapolation of results to the true population. We expect relative sampling efficiencies to be unaffected by *N*. There is no *a priori* reason to believe that one algorithm is better for small *N*, but not for large *N*. In terms of execution time required by each algorithm, extrapolation of execution times for

Table 7.2.1 Comparisons of the Test Population and the True Population

Variable	Population	Mean	Median	Minimum	Maximum	Std. Dev.
County Population	True	188881	41722	3764	941411	300786
	Test	189049	41722	3764	941411	299217
Number of Lanes	True	2.0416	2	1	10	0.3346
	Test	2.0350	2	1	6	0.2908
State/ Nonstate	True	0.0192	-	-	-	0.1373
	Test	0.0172	-	-	-	0.1300
Rural/ Urban	True	0.6462	-	-	-	0.4782
	Test	0.6442	-	-	-	0.4788

the test population to those needed in conjunction with the actual population is straightforward.

7.3 The Rank-1 Exchange Algorithm (Algorithm 5.1)

This is a single-point exchange algorithm. The modified Galil-Kiefer procedure (Section 5.5) is used to compute a starting design. The rank-1 exchange algorithm is then used to improve this starting design. To evaluate this algorithm in more general situations, we use the un-modified rank-1 exchange algorithm instead of the special version discussed in Section 5.8. The performance of the rank-1 exchange for categorical regressors can be estimated from the performance of the original algorithm.

As noted above, this algorithm is used in this chapter mainly as a benchmark for evaluating other algorithms because it will generate the best design among the five algorithms included in the study. The empirical study conducted in Section 5.6 indicated that this algorithm generates highly efficient designs. Therefore, the more important issue about this algorithm is whether it is applicable in very large populations.

From algorithm 5.1 in Section 5.4, it is obvious that most of the computation $(\frac{N\text{-}n}{N})$ is spent in the optimization of $\Delta^{+}(x)$ which is linear in N. We can therefore conjecture that the execution time of the rank-1 exchange algorithm is linear in N. We conducted an empirical study using another test population twice as large as the original test population. From this test population of 10,000 road sections, an optimal sample of size 151 was selected using the rank-1 exchange. From Table 7.3.1, we can see that the execution time for the large test population also nearly doubles that for the original test population.

Table 7.3.1 Relationship Between Execution Time and Problem Size: Rank-1 Exchange Algorithm (n = 151 in each case)

Problem Size	Rank-1 Exchange	
	V(s')	Execution time
5,000	0.0773	293.68
10,000	0.0825	577.89

Extrapolating these results to the entire population of 108,329 road sections, we estimated that the rank-1 exchange algorithm would require roughly 100 minutes to generate the optimal sample. However, if the special version of the rank-1 exchange algorithm discussed in Section 5.8 were implemented, there will be 3,480 distinct data groups. Since the problem size (N) is now only 3.2% of the original size, we estimate that it will take the modified rank-1 exchange algorithm roughly 4 minutes to find an optimal solution. With this modification, the rank-1 exchange algorithm becomes a very fast and efficient approach.

7.4 Resampling Based on Repeated Application of the Rank-1 Exchange Algorithm (Algorithm 5.2)

This algorithm selects a subpopulation from the test population for which an optimal sample is then selected. The process is repeated r times and the final solution is the sample with the smallest average variance of prediction.

To study the effectiveness of this algorithm, we selected 500, 1000, and 2000 points from the test population and treated them as subpopulations. After each subpopulation was selected, we used the rank-1 exchange algorithm to select the sample that minimizes $V(s')$ from the subpopulation . The rationale for the resampling approach is to optimize within smaller subpopulations so as to reduce the number of evaluations of Δ^+ required in the augmentation stage. Determination of the subpopulation size N_S, however, can be difficult and problem-dependent. If N_S is too small, the chance of obtaining a suboptimal solution increases; if N_S is close to N, the reductions in computer time are not realized.

Another issue in this approach is the selection of r (number of replications.) Again, the best value of r appears to be problem-dependent. It obviously depends on the choice of N_S also. (For instance, when $N_S = N$, the resampling approach becomes the original rank-1 exchange algorithm and replication becomes unnecessary.) With no general guidelines available about selecting the values of N_S and r, we use $r = 20$ for $N_S = 500$, 1000, and 2000 to obtain information on the distribution of $V(s')$. Numerical results are summarized in Table 7.4.1 and Figure 7.4.1. From these results, it appears that this algorithm is unstable when the subpopulation is small. However, when $N_S = 2{,}000$, this approach generates fairly consistent results, as indicated by the small standard deviation of $V(s')$. The designs generated using $N_S = 2{,}000$ have an average relative sampling efficiency of

91.4%, ranging from 88.5% to 94.7%. Therefore, when $N_S = 2{,}000$, it appears that we can use $r = 1$. If so, the computer time needed to generate an optimal sample using this algorithm would be about 125.68 seconds (2513.56/20.) This compares to 293.68 seconds required by the original rank-1 exchange algorithm. From Figure 7.4.2, it also appears that the execution time for this algorithm is linear in N_S. If this execution time is extrapolated to the true population, this algorithm would require about 45 minutes to generate an optimal sample.

Table 7.4.1 Relationships Between N_s, $V(s')$, and Execution Time Using the Resampling Approach (N = 5000, n = 151, r = 20)

N_s	Maximum V(s')	Minimum V(s')	Average V(s')	Standard Deviation of V(s')	Execution Time* (Seconds)
500	0.1490	0.0906	0.1043	0.0183	516.54
1,000	0.1022	0.0847	0.0896	0.0038	1070.78
2,000	0.0873	0.0816	0.0845	0.0015	2513.56

* Run on Cray-XMP

Figure 7.4.1 Relationships Between $V(s')$ and Size of Subpopulation Using the Resampling Approach (N = 5000, n = 151, r = 20)

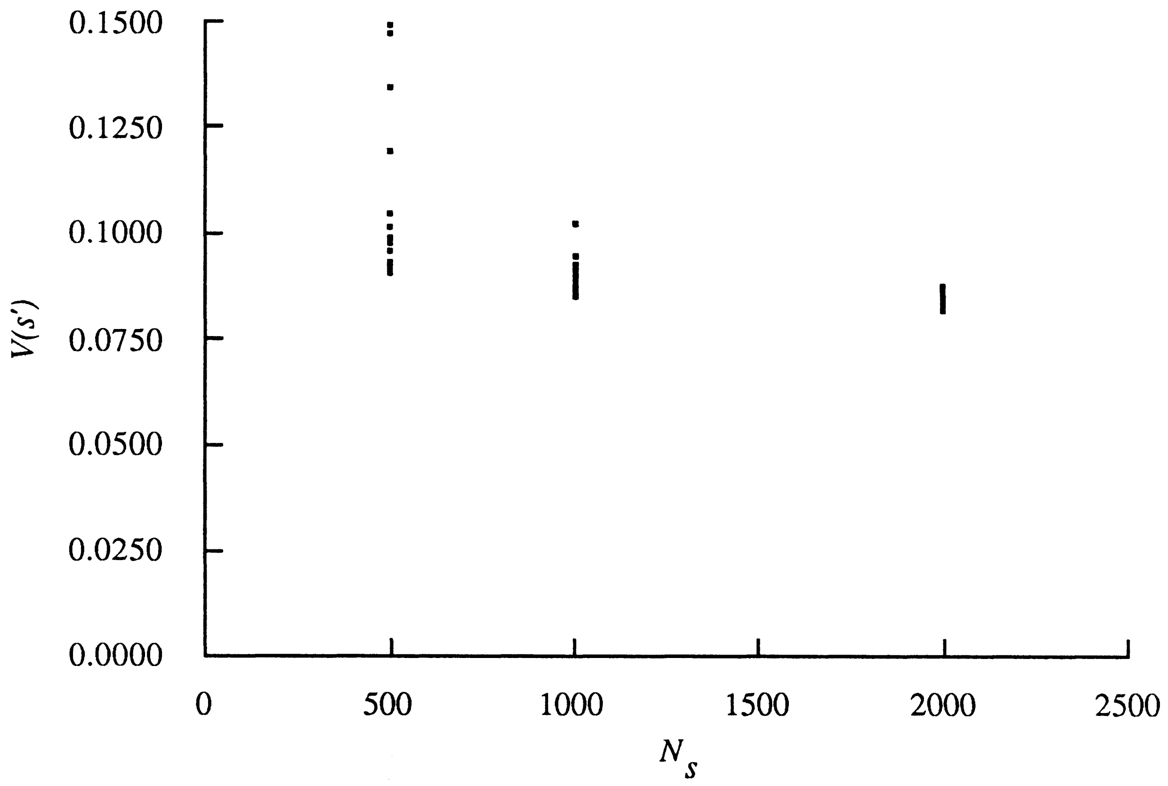
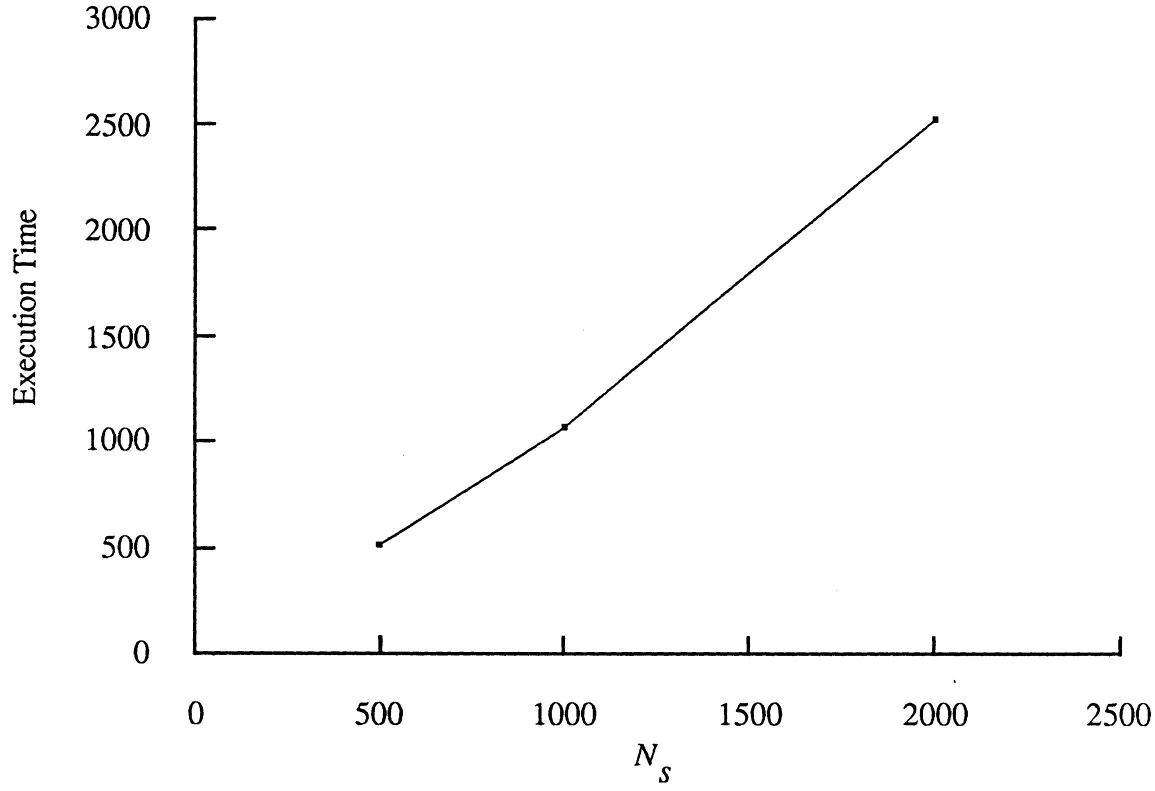


Figure 7.4.2 Relationships Between Execution Time and Size of Subpopulation Using the Resampling Approach (N = 5000, n = 151, r = 20)



7.5 Resampling Based on Repeated Simple Random Sampling (Algorithm 5.3)

To use this algorithm, we simply select a simple random sample from the population and compute the the corresponding $V(s')$. This process is repeated r times and the sample with the smallest $V(s')$ is chosen as the final solution.

To study the effectiveness of this algorithm, we selected a simple random sample of size 151 r times, where r range from 100 to 50,000. The smallest $V(s')$ was recorded for each value of r. The results are summarized in Table 7.5.1. From Figures 7.5.1 and 7.5.2, it can be seen that the computer time increases linearly with r while $V(s')$ stays

Table 7.5.1 Relationships between Minimum Average Variance of Prediction, Execution Time and Number of Iterations Using Simple Random Sampling

Number of Iterations	Minimum Average Variance of Prediction	Execution Time* (Seconds)
100	0.0973	3.25
200	0.0970	6.28
300	0.0970	6.35
400	0.0970	11.80
500	0.0970	14.77
1,000	0.0967	25.45
2,000	0.0959	48.03
5,000	0.0951	109.76
6,000	0.0951	130.72
7,000	0.0951	167.08
8,000	0.0951	184.10
10,000	0.0951	186.45
20,000	0.0948	439.78
50,000	0.0943	1074.70

* Run on Cray-XMP

relatively unchanged. Execution times and $V(s')$ for the rank-1 exchange algorithm and two-stage sampling are also included in these two figures for comparison. After $r > 200$, algorithm 5.3 was completely dominated by two-stage sampling which generated better designs in about the same time. After 10,000 iterations, this approach used more computer time than the rank-1 exchange algorithm while the design generated was much less satisfactory.

Figure 7.5.1 Execution-Time Profile for Repeated Simple Random Sampling. Execution Times for the Rank-1 Exchange Algorithm and for Two-stage Sampling are shown for Comparison. (N = 5000, n = 151)

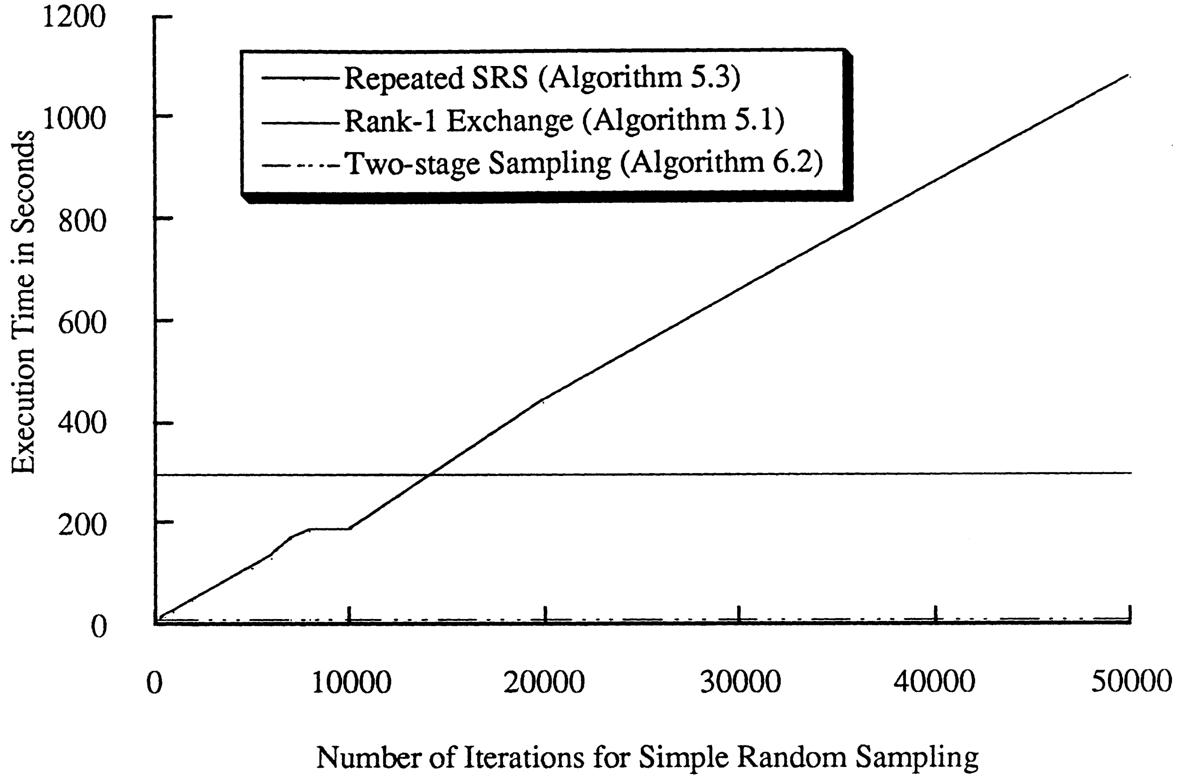
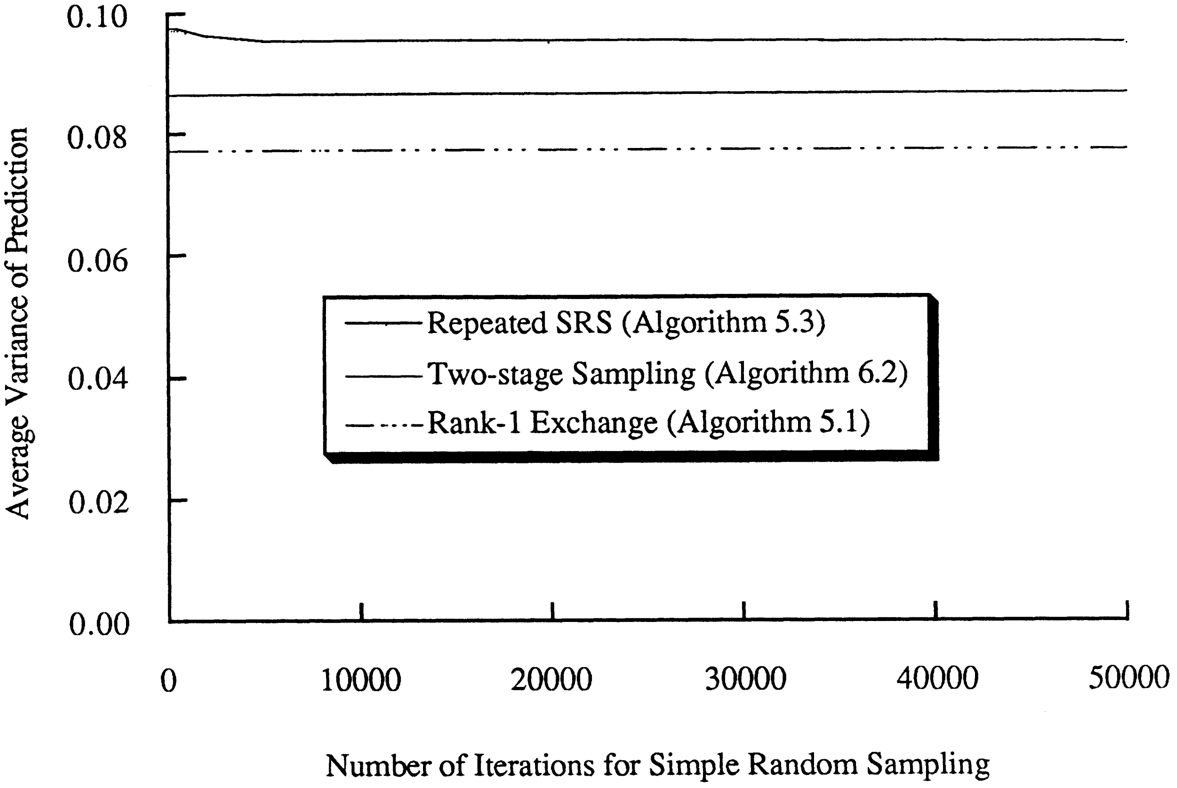


Figure 7.5.2 **Profile of Average Prediction Variance for Repeated Simple Random Sampling. Average Prediction Variances for the Rank-1 Exchange Algorithm and for Two-stage Sampling (*k* = 10 clusters) are shown for Comparison. (*N* = 5000, *n* = 151)**



The expected execution time for the true population will depend on the sampling efficiency required. Judging from the test population results, it appears $V(s')$ does not decrease after r is greater than approximately 5,000. The relative sampling efficiency at this stage was about 81% compared to the rank-1 exchange algorithm and the execution time was 109.76 seconds. The execution time needed for the true population will be about 40 minutes. However, we expect the sampling efficiency of this approach to be lower for true population because it will become more and more difficult to "run into" a good design when N becomes larger.

7.6 Two-stage Sampling with Random Selection Within Clusters (Algorithm 6.2)

Under this approach, the population is first divided into similar clusters. Then approximate design techniques are used to calculate optimal weights for each cluster. Finally, points are selected randomly from each cluster based on the round-off numbers from these optimal weights.

To see the effect of N on the sampling efficiency and execution time of this algorithm, we again apply it to a test population containing 10,000 road sections. The results, shown in Table 7.6.1, indicate that:

(1) the execution time needed for this algorithm is roughly linear in N, and

(2) the sampling efficiency of this algorithm remains about 89% regardless of N.

Extending the execution time to the true population, we estimated that this algorithm would require about 2.5 minutes to generate an optimal sample which is by far the fastest algorithm.

Table 7.6.1 Relationship Between Execution Time and Problem Size: Rank-1 Exchange (Algorithm 5.1) and Two-stage Sampling (Algorithm 6.2, *k* = 10 clusters) (*n* = 151 in each case)

Problem Size	Rank-1 Exchange		Two-stage Sampling	
	V(s')	Execution time	V(s')	Execution time
5,000	0.0773	293.68	0.0865	7.47
10,000	0.0825	577.89	0.0923	15.31

7.7 Two-stage Sampling with Optimal Selection Within Clusters (Algorithm 6.3)

This algorithm is similar to algorithm 6.2 except that the rank-1 exchange algorithm is applied within clusters to improve the design. The rationale for this algorithm is to replace "bad" points with "good" points within a cluster in the event that the population units do not cluster well (i.e., road sections within clusters are not homogeneous.)

Because of the extra step involved, this algorithm takes much longer to generate an optimal sample than algorithm 6.2. Using the test population of 5,000 sections, the computer time needed for this algorithm was about 20 times as long as that for algorithm 6.2 (see Table 7.7.1.) The improvement in $V(s')$, however, was minimal. This could be a result where clusters were fairly homogeneous and rank-1 exchange among similar points within clusters offered little improvements.

Another interesting result from the test population is that choosing an appropriate number of clusters offered more improvement than optimal selection within clusters. The empirical study conducted in Section 6.6.1 indicated that $V(s')$ can sometimes be greatly affected by the number of clusters used. A similar study was conducted here using the test population. Results in Table 7.7.2 and Figure 7.7.1 showed that $V(s')$ with different

Table 7.7.1 $V(s')$ and Execution Time of Random Selection and Optimal Selection of Two-stage Sampling (k = 10 clusters) (N = 5000, n = 151)

	$V(s')$	Execution Time*
Random Selection	0.0865	7.47
Optimal Selection	0.0839	160.41

* Run on Cray-XMP

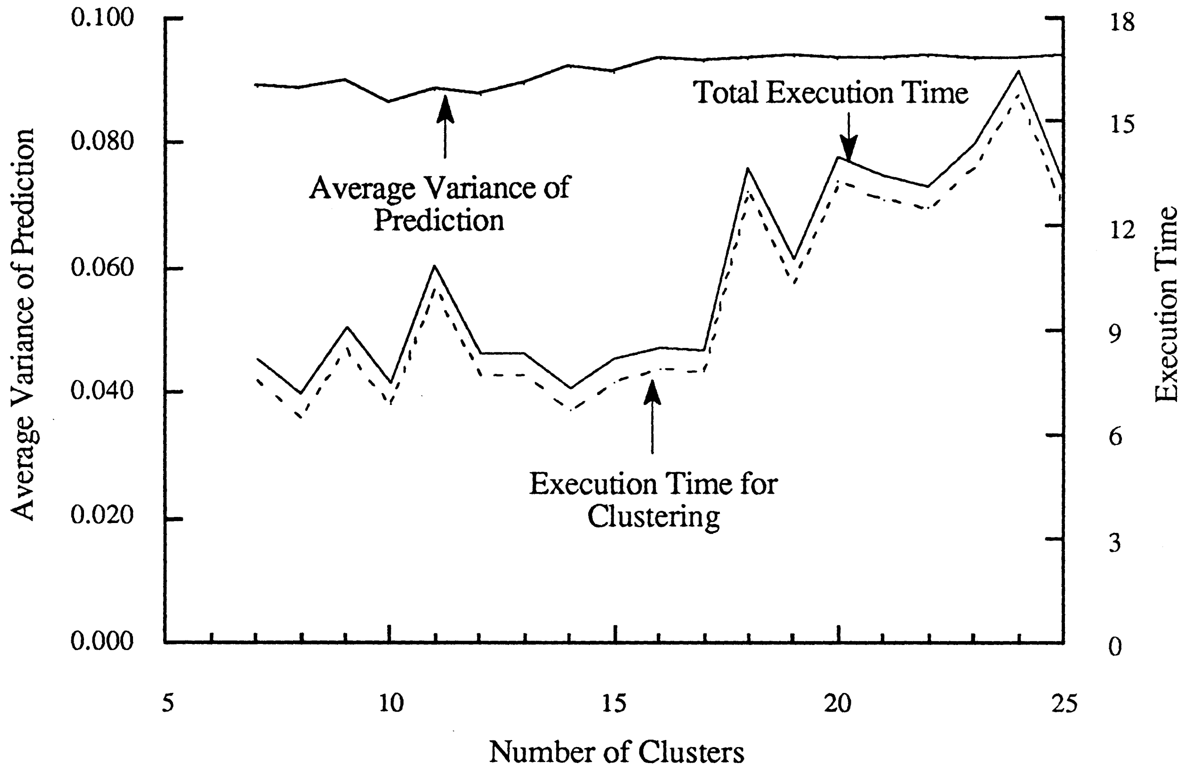
number of clusters can differ by as much as .0072. This is equivalent to a difference of more than 9% in sampling efficiency. Optimal exchange within clusters, by comparison, offers only about 3% improvement in sampling efficiency in the cases we tested. Since it takes much less computer time to cluster points than to perform exchanges, it may be more beneficial to spend computer time finding the optimal number of clusters rather than performing exchanges within clusters. Once the optimal number of clusters (the one with the smallest average prediction variance) is determined, random selection within clusters generates highly efficient designs in a fraction of time needed for optimal selection. Identifying the optimal number of clusters may also offer some insight of the true cluster structure of the population. This is because the performance of algorithm 6.2 is affected by how well the population units clustered. In this example, we might conclude that the population is best clustered into 10 groups.

Table 7.7.2 Relationship Between Average Variance of Prediction, Execution time, and Number of Clusters for Algorithm 6.2 - Random Selection within Clusters

Number of Clusters	$V(s')$	Execution Time* for Clustering (seconds)	Total Execution Time* (seconds)
7	0.0887	7.54	8.20
8	0.0887	6.47	7.16
9	0.0896	8.44	9.12
10	0.0865	6.82	7.47
11	0.0884	10.20	10.85
12	0.0874	7.67	8.33
13	0.0893	7.71	8.35
14	0.0919	6.67	7.33
15	0.0912	7.49	8.16
16	0.0931	7.86	8.50
17	0.0928	7.78	8.43
18	0.0931	12.96	13.63
19	0.0936	10.31	11.00
20	0.0933	13.25	13.97
21	0.0933	12.77	13.43
22	0.0937	12.47	13.11
23	0.0933	13.67	14.32
24	0.0934	15.78	16.43
25	0.0935	12.48	13.13
Total Execution Time (All Trials): 200.92 Seconds			

* Run on Cray-XMP

Figure 7.7.1 Relationship Between Average Variance of Prediction, Execution time, and Number of Clusters for Algorithm 6.2 - Random Selection within Clusters



7.8 Recommendations

The execution times and $V(s')$ of the five algorithms evaluated in this chapter using the test population are summarized in Table 7.8.1 and Figure 7.8.1. From Figure 7.8.1, we can see that the rank-1 exchange algorithm and two-stage sampling with random selection within clusters represent the two extremes of the spectrum. The rank-1 exchange algorithm is highly efficient but slow while two-stage sampling with random selection is fast but less efficient. For the highway planning problem addressed in this report, we recommend the special version of the rank-1 exchange algorithm because of the categorical

Table 7.8.1 Execution Times and V(s') of the Five Algorithms Evaluated

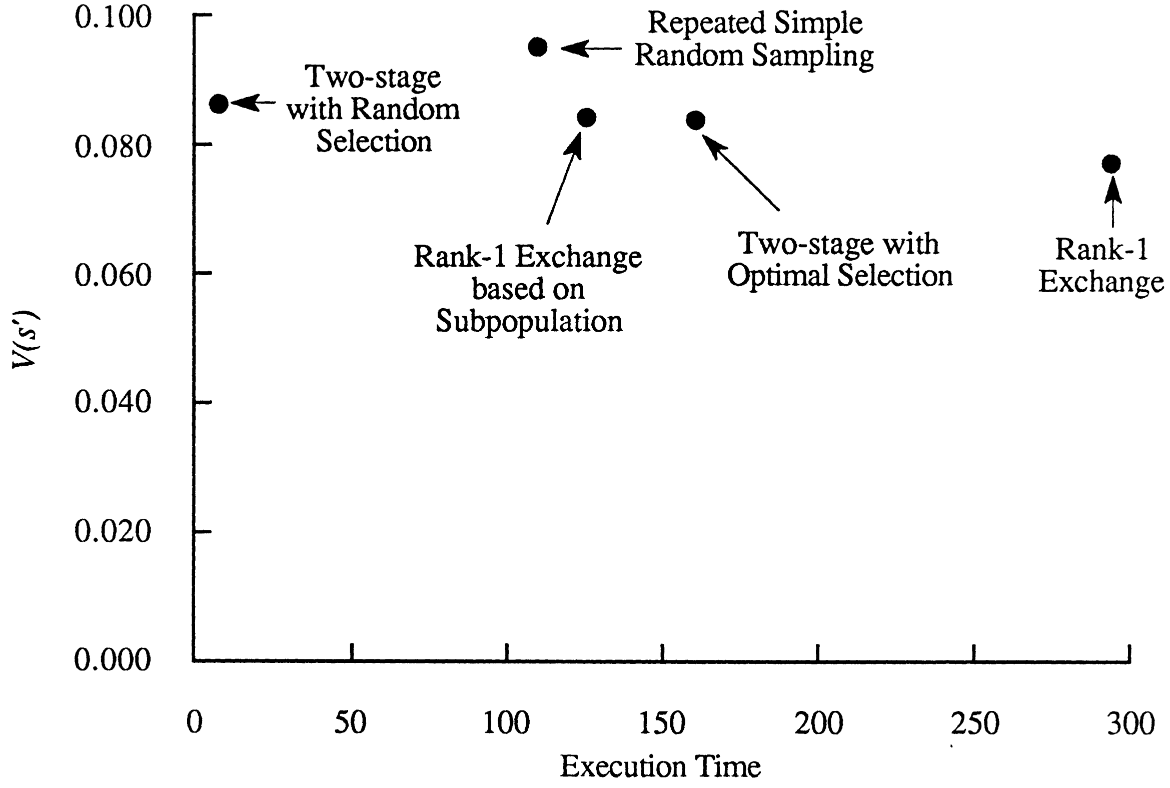
Algorithm	Execution Time	$V(s')$
Rank-1 Exchange Algorithm	293.68	0.0773
Repeated Application of Rank-1 Exch. on N_s	125.68	0.0845
Repeated Simple Random Sampling	109.76	0.0951
Two-stage Sampling with Random Selection*	7.47	0.0865
Two-stage Sampling with Optimal Selection*	160.41	0.0839

* k = 10 clusters

nature of the population units. In more general problems, the trade-off again is between computing resource and sampling efficiency. With the advent of powerful work stations, computing is becoming less of a problem. As a result, the rank-1 exchange algorithm will probably be the preferred approach in the future.

Of interest is the performance of the rank-1 exchange algorithm and the two-stage sampling approach relative to simple random sampling. We conducted an empirical study similar to the one in Section 6.6.2. The test population contains the same 5,000 road sections used elsewhere in this chapter. In the first stage of the two-stage sampling algorithm, this test population was partitioned into 28 clusters. We again varied the number of simulated sample to see its effect on the range of $V(s')$. Numerical results are summarized in Table 7.8.2. From this table, it is clear that the two-stage sampling algorithm generates more homogeneous designs than simple random sampling . The difference in the average variance of prediction between the best and the worst designs is very small when two-stage sampling algorithm is used. Furthermore, the average relative efficiency of simple random sampling is less than 5% (in comparison with the two-stage

Figure 7.8.1 Execution Times and V(s') of the Five Algorithms Evaluated



approach.) This means we are likely to select a very bad design in this larger population if we rely on a single trial of simple random sampling.

When compared to the best design generated by the rank-1 exchange algorithm, designs generated using algorithm 6.2 have a relative sampling efficiency of 82% approximately. If the number of clusters used were 10, the relative sampling efficiency of designs generated using algorithm 6.2 can be improved to 89% for this particular test population (Section 7.6.)

Table 7.8.2 Comparison of $V(s')$ for the Rank-1 Exchange Algorithm, Two-stage Sampling Approach, and Simple Random Sampling ($N = 5,000$, $n = 151$)

Number of Simulated Samples	Simple Random Sampling			Clustering* + Optimal Weighting/ SRS within cluster			Efficiency of SRS
	Max.	Min.	Avg.	Max.	Min.	Avg.	
1000	599.280	.097	3.230	.095	.093	.094	.0290
2000	599.275	.096	3.263	.095	.093	.094	.0413
5000	1055.100	.095	2.587	.095	.093	.094	.0362
Optimal Sampling Via Rank-1 Exchange Algorithm: .077							

* The test population containing 5,000 road sections was clustered into 28 clusters.

7.9 Implementation Issues

Since ATR sites are semi-permanent installations, the main use of the rank-1 exchange algorithm should be in the augmentation of a sample design. In other words, the rank-1 exchange algorithm should be used to locate new ATR installations. We simply compute $\Delta^+(x)$ for all potential ATR installation sites and select the site having the maximum $\Delta^+(x)$. The set of potential sites can be selected by traffic engineers to reduce the problem size and at the same time give them more control as to where to install the new ATRs.

The two-stage sampling approach can also be modified to give traffic engineers more control over sampling. For instance, instead of random selection within clusters, a judgmental sample can be selected by traffic engineers (still in accordance with the optimal weights.)

In the future, as video cameras gradually replace automatic traffic recorders, relocation of monitoring sites will become easier. When that happens, either the rank-1 exchange algorithm or the two-stage sampling approach can be used periodically to revise the set of optimal monitoring sites. This will help ensure the highest possible precision for AADT estimation at specific sites.

CHAPTER 8
DISCUSSION AND EXTENSION

This chapter considers extensions of this research. These include: model robustness studies, the scheduling of mobile traffic counters, more empirical studies of the two major numerical algorithms developed in this report, and more in-depth studies of the spatial correlation problem.

8.1 Model Robustness

In this research, we did not emphasize model robustness issues for the following two reasons: (1) The main purpose of this project was to develop fast numerical algorithms to find optimal samples from very large populations, and (2) we feel the regression model accurately and adequately describes the relationship between traffic-volume data and other predictors, especially given the limitations of data availability. Model robustness is more typically a concern when a priori knowledge of the model is subjective. In this problem, the regression model can be estimated in advance of sampling and continually updated.

However, uncertainty about the superpopulation model has been one of the major criticisms of model-based sampling techniques. While many model-based sampling researchers have argued that the selection of the sample should not be left to a chance mechanism when information about the population of interest is available, design-based samplers have held that improper reliance on an incorrect model can seriously degrade the ensuing inference. A model robustness study--in which a sensitivity analysis of the model-based sampling technique can be conducted--is a natural extension of the current research. Such a study would consider each of the following:

(1) the form of the regression (superpopulation) model, and

(2) the form of the variance function of the errors.

Researchers have proposed various approaches to deal with the problem of model specification uncertainty in model-based sampling. (For a general discussion of model specification uncertainty, see Benson and Barry [1982]). These approaches are categorized and discussed briefly below:

(1) Balanced sampling approach (Royall and Herson, [1973]): In this approach, a purposive sample is chosen so that the first *j*-th sample moments of the control variables match population moments of the same variables. According to Royall and Herson, this procedure will protect against model misspecification up to polynomial model of *j*-th degree. This approach, however, may lead to inefficient sampling. Therefore, some compromises between optimal sampling and balanced sampling might be of interest.

(2) Asymptotic Design Unbiasedness (ADU) approach: Brewer's approach [1979] and other extensions such as Isaki and Fuller's [1982] and Wright's [1983] are in this category. This approach is somewhat similar to the classical "Neyman allocation" approach in which the inclusion probability of units in a stratum is proportional to the standard deviation of the stratum.

(3) Model-robust designs: In this approach, an experimenter assumes the "true" model is unknown, but is an element in a known family of models denoted by *F*. Optimality criteria are measured with respect to the family of models as a whole. Thibodeau [1977] considered designs that maximize the minimum *G*-efficiency over *F* while Läuter [1974] proposed maximizing the average. Cook and Nachtsheim [1982] considered model robust *L*-optimal designs in which they defined a design to be $\overline{L}$ optimal if it minimizes the average inefficiency

$$L(\xi) = \int_I E_i^{-1}(\xi) d\beta(i)$$

where

$$E_i(\xi) = \frac{L(\mathbf{D}_i(\xi_i))}{L(\mathbf{D}_i(\xi))}$$

is the L-efficiency when the i-th model is true, **D** denotes the dispersion matrix of the design, I is the indices for models in F and β reflects an experimenter's belief about the likelihood of each model in F.

8.2 Scheduling Mobile Traffic Counters

The major practical problem addressed in this report was how to optimally locate Automatic Traffic Recorders throughout a roadway system. We formulated this problem as a one dimensional sampling problem in the sense that we are only concerned about where to take measurements. A different and more complicated problem is the placement of the mobile traffic counters. As explained in Chapter 2, the short-duration counts of mobile traffic counters provide the majority of the sample data used in AADT estimation. Because these counters are mobile units, the sampling design has to consider not only the location but also the timing of traffic-monitoring. Optimal placement of these mobile counters is a much more complicated problem than the optimal placement of ATRs because the total number of possible arrangement increases dramatically. If this sampling problem can be solved with statistical and/or mathematical programming techniques, the precision of AADT estimates can be further improved. Benson, Pisharody, and Yeldan [1986] suggested approaches for addressing the timing problem.

8.3 More Empirical Studies about the Rank-1 Exchange and Two-stage Sampling Algorithms

In this report, we compared the sampling efficiency of the optimal sampling strategies to that of simple random sampling. But simple random sampling is only one of numerous design-based sampling techniques. In situations where information about the population of interest is available, other design-based sampling technique such as stratified sampling or unequal probability sampling techniques are more likely to be used. It is therefore of interest to find out how the performance of optimal sampling techniques compare to that of other design-based sampling techniques. Although the studies will be empirical in nature and cannot be generalized to other problems, the result can at least give us additional assessments of the performance of the algorithms proposed in this research.

8.4 More In-depth Study of the Spatial Correlation Problem

As pointed out previously, one of the unique problems in this research is that the AADTs of neighboring road sections are not independent. In Chapter 4, we suggested some possible approaches in dealing with the spatial correlation between population units. The solution proposed herein is admittedly heuristic in that the potential spatial correlations are not explicitly considered. More systematic approaches relating to kriging or other spatial sampling techniques could be considered.

APPENDIX A

Proofs of Propositions 5.4.1 and 5.4.2

Proofs of Propositions 5.4.1 and 5.4.2

The average variance of prediction over the nonsampled set can be expressed as follows:

$$V(s') = \frac{1}{N-n}\left[\sum_{x_i \in s'} \sigma_i^2 + \sum_{x_i \in s'} f(x_i)^T M^{-1}(s) f(x_i)\right]$$

where

$M^{-1}(s) = (X_S{}^T V^{-1} X_S)^{-1}$ is the dispersion matrix of the old design. $M^{-1}(s)$ can also be expressed as:

$$\left[\sum_{x_i \in s} \frac{1}{\sigma_i^2} f(x_i) f(x_i)^T\right]^{-1} = \left[\sum_{x_i \in s} g(x_i) g(x_i)^T\right]^{-1}$$

where

$$g(x_i) = \frac{1}{\sqrt{\sigma_i^2}} f(x_i)$$

After x_k is added to the sampled set, the new non-sampled set becomes

$$s'_{new} = s' - x_k.$$

Now the new average variance of prediction can be expressed as:

$$V(s'_{new}) = \frac{1}{N-n-1}\left[\sum_{x_i \in s'_{new}} \sigma_i^2 + \sum_{x_i \in s'_{new}} f(x_i)^T M^{-1}(s_{new}) f(x_i)\right].$$

$M^{-1}(s_{new})$ can be calculated from $M^{-1}(s)$ by the following matrix update formula (Fedorov, [1972]):

$$\begin{aligned} M^{-1}(s_{new}) &= \left[M(s) + g(x_k) g(x_k)^T\right]^{-1} \\ &= M^{-1}(s) - M^{-1}(s) g(x_k) g(x_k)^T \left[I + M^{-1}(s) g(x_k) g(x_k)^T\right]^{-1} M^{-1}(s) \end{aligned}$$

But

$$\left[I + M^{-1}(s) g(x_k) g(x_k)^T\right]^{-1} = I - M^{-1}(s) g(x_k) \left[1 + g(x_k)^T M^{-1}(s) g(x_k)\right]^{-1} g(x_k)^T$$

and

$$\left[1+g(x_k)^T M^{-1}(s)g(x_k)\right]^{-1} = \frac{1}{1+\frac{1}{\sigma_k^2}f(x_k)^T M^{-1}(s)f(x_k)} = \frac{1}{1+\frac{v_{kk}}{\sigma_k^2}} = \frac{\sigma_k^2}{\sigma_k^2+v_{kk}},$$

where $v_{kk} = f(x_k)^T M^{-1}(s)f(x_k)$.

Let $C_k = \frac{\sigma_k^2}{\sigma_k^2+v_{kk}}$. Then

$$\begin{aligned}
M^{-1}(s_{new}) &= M^{-1}(s) - M^{-1}(s)g(x_k)g(x_k)^T\left[I - M^{-1}(s)g(x_k)C_k g(x_k)^T\right]M^{-1}(s) \\
&= M^{-1}(s) - \frac{1}{\sigma_k^2}M^{-1}(s)f(x_k)f(x_k)^T M^{-1}(s) \\
&\quad + \frac{C_k}{\left(\sigma_k^2\right)^2}M^{-1}(s)f(x_k)f(x_k)^T M^{-1}(s)f(x_k)f(x_k)^T M^{-1}(s) \\
&= M^{-1}(s) - \frac{1}{\sigma_k^2}M^{-1}(s)f(x_k)f(x_k)^T M^{-1}(s) \\
&\quad + \frac{C_k}{\left(\sigma_k^2\right)^2}M^{-1}(s)f(x_k)v_{kk}f(x_k)^T M^{-1}(s) \\
&= M^{-1}(s) - \left(\frac{1}{\sigma_k^2} - \frac{C_k v_{kk}}{\left(\sigma_k^2\right)^2}\right)M^{-1}(s)f(x_k)f(x_k)^T M^{-1}(s) \\
&= M^{-1}(s) - \left(\frac{1}{\sigma_k^2+v_{kk}}\right)M^{-1}(s)f(x_k)f(x_k)^T M^{-1}(s).
\end{aligned}$$

Therefore,

$$\begin{aligned}
V(s'_{new}) &= \frac{1}{N-n-1}\left\{\left(\sum_{x_i\in s'}\sigma_i^2 - \sigma_k^2\right) + \left[\sum_{x_i\in s'} f(x_i)^T M^{-1}(s_{new})f(x_i) - f(x_k)^T M^{-1}(s_{new})f(x_k)\right]\right]\right\} \\
&= \frac{1}{N-n-1}\left\{\left(\sum_{x_i\in s'}\sigma_i^2 - \sigma_k^2\right)\right\}
\end{aligned}$$

$$+\frac{1}{N-n-1}\left\{\sum_{x_i\in s'} f(x_i)^T\left[M^{-1}(s)-\left(\frac{1}{\sigma_k^2+v_{kk}}\right)M^{-1}(s)f(x_k)f(x_k)^T M^{-1}(s)\right]f(x_i)\right\}$$

$$-\frac{1}{N-n-1}\left\{f(x_k)^T\left[M^{-1}(s)-\left(\frac{1}{\sigma_k^2+v_{kk}}\right)M^{-1}(s)f(x_k)f(x_k)^T M^{-1}(s)\right]f(x_k)\right\}$$

$$=\frac{1}{N-n-1}\left(\sum_{x_i\in s'}\sigma_i^2+\sum_{x_i\in s'}f(x_i)^T M^{-1}(s)f(x_i)\right)$$

$$-\frac{1}{N-n-1}\left[\sigma_k{}^2+\frac{1}{\sigma_k^2+v_{kk}}\sum_{x_i\in s'}v_{ik}{}^2-v_{kk}+\frac{1}{\sigma_k^2+v_{kk}}v_{kk}{}^2\right]$$

$$=\frac{1}{N-n-1}\left\{(N-n)V(s')-\Delta^+(x_k)\right\}.$$

Proof of Proposition 5.4.2.

$$V(s') = \frac{1}{N-n}\left[\sum_{x_i \in s'} \sigma_i^2 + \sum_{x_i \in s'} f(x_i)^T M^{-1}(s) f(x_i)\right]$$

Now suppose x_h is deleted from the sampled set, denote the new non-sampled set by s'_{new} where $s'_{new} = s' + x_h$.

$$V(s'_{new}) = \frac{1}{N-n+1}\left[\sum_{x_i \in s'_{new}} \sigma_i^2 + \sum_{x_i \in s'_{new}} f(x_i)^T M^{-1}(s_{new}) f(x_i)\right]$$

where

$$\begin{aligned} M^{-1}(s_{new}) &= \left[M(s) - g(x_h)g(x_h)^T\right]^{-1} \\ &= M^{-1}(s) + M^{-1}(s)g(x_h)g(x_h)^T\left[I - M^{-1}(s)g(x_h)g(x_h)^T\right]^{-1} M^{-1}(s). \end{aligned}$$

Again,

$$\left[I - M^{-1}(s)g(x_h)g(x_h)^T\right]^{-1} = I + M^{-1}(s)g(x_h)\left[1 - g(x_h)^T M^{-1}(s)g(x_h)\right]^{-1} g(x_h)^T,$$

and

$$\begin{aligned} \left[1 - g(x_h)^T M^{-1}(s)g(x_h)\right]^{-1} &= \frac{1}{1 - \frac{1}{\sigma_h^2} f(x_h)^T M^{-1}(s) f(x_h)} \\ &= \frac{1}{1 - \frac{v_{hh}}{\sigma_h^2}} = \frac{\sigma_h^2}{\sigma_h^2 - v_{hh}} \end{aligned}$$

where $v_{hh} = f(x_h)^T M^{-1}(s) f(x_h)$.

Let $C_h = \dfrac{\sigma_h^2}{\sigma_h^2 - v_{hh}}$

then

$$\begin{aligned} M^{-1}(s_{new}) &= M^{-1}(s) + M^{-1}(s)g(x_h)g(x_h)^T\left[I + M^{-1}(s)g(x_h)C_h g(x_h)^T\right]M^{-1}(s) \\ &= M^{-1}(s) + \frac{1}{\sigma_h^2} M^{-1}(s) f(x_h) f(x_h)^T M^{-1}(s) \end{aligned}$$

$$+\frac{C_h}{\left(\sigma_h^2\right)^2} M^{-1}(s) f(x_h) f(x_h)^T M^{-1}(s) f(x_h) f(x_h)^T M^{-1}(s)$$

$$= M^{-1}(s) + \frac{1}{\sigma_h^2} M^{-1}(s) f(x_h) f(x_h)^T M^{-1}(s)$$

$$+\frac{C_h}{\left(\sigma_h^2\right)^2} M^{-1}(s) f(x_h) v_{hh} f(x_h)^T M^{-1}(s)$$

$$= M^{-1}(s) + \left(\frac{1}{\sigma_h^2} + \frac{C_h v_{hh}}{\left(\sigma_h^2\right)^2}\right) M^{-1}(s) f(x_h) f(x_h)^T M^{-1}(s)$$

$$= M^{-1}(s) + \left(\frac{1}{\sigma_h^2 - v_{hh}}\right) M^{-1}(s) f(x_h) f(x_h)^T M^{-1}(s)$$

$$V\left(s'_{new}\right) = \frac{1}{N-n+1}\left[\sum_{x_i \in s'_{new}} \sigma_i^{\,2} + \sum_{x_i \in s'_{new}} f(x_i)^T M^{-1}\left(s_{new}\right) f(x_i)\right]$$

$$= \frac{1}{N-n+1}\left(\sum_{x_i \in s'} \sigma_i^{\,2} + \sigma_h^{\,2}\right)$$

$$+\frac{1}{N-n+1}\left[f(x_h)^T M^{-1}\left(s_{new}\right) f(x_h) + \sum_{x_i \in s'} f(x_i)^T M^{-1}\left(s_{new}\right) f(x_i)\right]$$

$$= \frac{1}{N-n+1}\left(\sum_{x_i \in s'} \sigma_i^{\,2} + \sum_{x_i \in s'} f(x_i)^T M^{-1}(s) f(x_i)\right)$$

$$+\frac{1}{N-n+1}\left[\sigma_h^{\,2} + \frac{1}{\sigma_h^2 - v_{hh}}\left[v_{hh}\sigma_h^2 + \sum_{i \in s'} v_{ih}^{\,2}\right]\right]$$

$$= \frac{1}{N-n+1}\left[(N-n)V(s') + \sigma_h^{\,2} + \frac{1}{\sigma_h^2 - v_{hh}}\left[v_{hh}\sigma_h^2 + \sum_{i \in s'} v_{ih}^{\,2}\right]\right]$$

$$= \frac{1}{N-n+1}\left[(N-n)V(s') + \Delta^{-}(x_h)\right].$$

APPENDIX B

Proof of Proposition 6.2.1

Proof of Proposition 6.2.1

$$\frac{\partial}{\partial\alpha}V\left(\xi_{j+1}\right)\Big|_{\alpha=0}$$

$$= TR\left[M^{-1}\left(\xi_j\right)W_{\xi_0}\right] - F\left[\sigma_{\bar{x}^*}^{-2}\phi\left(\bar{x}^*, \xi_i, W_{\xi_0} - W_{\xi_j}\right) + N\sigma_{\bar{x}^*}^2 + d\left(\bar{x}^*, \xi_j\right)\right] + NE_{\xi_j}\left[\sigma_{\bar{x}}^2\right]$$

where

$$\phi(x,\xi,W) = f^T(\bar{x})\left[M^{-1}(\xi)WM^{-1}(\xi)\right]f(\bar{x})$$
$$d(x,\xi) = f^T(\bar{x})M^{-1}(\xi)f(\bar{x})$$

and

$$W_\xi = \sum_{i=1}^{k} f(\bar{x})f(\bar{x})^T\xi(\bar{x}).$$

In order to prove this proposition, we need the following lemma.

Lemma 6.2.1:

$$\frac{\partial}{\partial\alpha}M^{-1}\left(\xi_{j+1}\right) = -M^{-1}\left(\xi_{j+1}\right)\left[M\left(\xi_{\bar{x}^*}\right) - M\left(\xi_j\right)\right]M^{-1}\left(\xi_{j+1}\right)$$

Proof.

From Fedorov [1972, p. 21],

$$\frac{\partial}{\partial\alpha}A^{-1} = -A^{-1}\left(\frac{\partial}{\partial\alpha}A\right)A^{-1}$$

where A is a square matrix and $|A| \neq 0$.

Let $A = M\left(\xi_{j+1}\right)$, we have

$$\frac{\partial}{\partial\alpha}M^{-1}\left(\xi_{j+1}\right) = -M^{-1}\left(\xi_{j+1}\right)\left(\frac{\partial}{\partial\alpha}M\left(\xi_{j+1}\right)\right)M^{-1}\left(\xi_{j+1}\right).$$

But,

$$\frac{\partial}{\partial\alpha}M\left(\xi_{j+1}\right) = \frac{\partial}{\partial\alpha}\left[(1-\alpha)M\left(\xi_j\right) + \alpha M\left(\xi_{\bar{x}^*}\right)\right] = M\left(\xi_{\bar{x}^*}\right) - M\left(\xi_j\right),$$

and the result follows.

<u>Proof of Proposition 6.2.1</u>:

Since

$$V(\xi)=\frac{1}{N-n}\left[N\left(E_{\xi_0}\left[\sigma_{\bar{x}}^2\right]-E_{\xi}\left[\sigma_{\bar{x}}^2\right]\right)+TR\left[M^{-1}(\xi)W_{\xi_0}\right]-TR\left[M^{-1}(\xi)W_{\xi}\right]\right],$$

we have

$$\frac{\partial}{\partial\alpha}V\left(\xi_{j+1}\right)=TR\left[\frac{\partial}{\partial\alpha}M^{-1}\left(\xi_{j+1}\right)W_{\xi_0}\right]-TR\left[\frac{\partial}{\partial\alpha}M^{-1}\left(\xi_{j+1}\right)W_{\xi_{j+1}}\right]-N\frac{\partial}{\partial\alpha}E_{\xi_{j+1}}\left[\sigma_{\bar{x}}^2\right]$$

since $E_{\xi_0}\left[\sigma_{\bar{x}}^2\right]$ is a constant with respect to α. We considered each of the three terms on the right in turn.

Term 1:

$$TR\left[\frac{\partial}{\partial\alpha}M^{-1}\left(\xi_{j+1}\right)W_{\xi_0}\right]=TR\left[\left\{-M^{-1}\left(\xi_{j+1}\right)\left[M\left(\xi_{\bar{x}^*}\right)-M\left(\xi_j\right)\right]M^{-1}\left(\xi_{j+1}\right)\right\}W_{\xi_0}\right]$$

Term 2:

$$-TR\left[\frac{\partial}{\partial\alpha}M^{-1}\left(\xi_{j+1}\right)W_{\xi_{j+1}}\right]=-TR\left[\frac{\partial}{\partial\alpha}M^{-1}\left(\xi_{j+1}\right)\sum_{i=1}^{k}f(\bar{x}_i)f^T(\bar{x}_i)\xi_{j+1}(\bar{x}_i)\right]$$

$$=-TR\left[\frac{\partial}{\partial\alpha}M^{-1}\left(\xi_{j+1}\right)\sum_{i=1}^{k}f(\bar{x}_i)f^T(\bar{x}_i)\left[(1-\alpha)\xi_j(\bar{x}_i)+\alpha\xi_{\bar{x}^*}(\bar{x}_i)\right]\right]$$

$$=-\frac{\partial}{\partial\alpha}\sum_{i=1}^{k}f^T(\bar{x}_i)M^{-1}\left(\xi_{j+1}\right)f(\bar{x}_i)(1-\alpha)\xi_j(\bar{x}_i)$$

$$-\frac{\partial}{\partial\alpha}\sum_{i=1}^{k}f^T(\bar{x}_i)M^{-1}\left(\xi_{j+1}\right)f(\bar{x}_i)\alpha\xi_{\bar{x}^*}(\bar{x}_i)$$

$$=-\sum_{i=1}^{k}f^T(\bar{x}_i)\left[\frac{\partial}{\partial\alpha}M^{-1}\left(\xi_{j+1}\right)\right]f(\bar{x}_i)(1-\alpha)\xi_j(\bar{x}_i)$$

$$-\sum_{i=1}^{k}f^T(\bar{x}_i)M^{-1}\left(\xi_{j+1}\right)f(\bar{x}_i)\left[\frac{\partial}{\partial\alpha}(1-\alpha)\xi_j(\bar{x}_i)\right]$$

$$-\sum_{i=1}^{k} f^T(\bar{x}_i)\left[\frac{\partial}{\partial\alpha}M^{-1}\left(\xi_{j+1}\right)\right]f^T(\bar{x}_i)\alpha\xi_{\bar{x}^*}(\bar{x}_i)$$

$$-\sum_{i=1}^{k} f^T(\bar{x}_i)M^{-1}\left(\xi_{j+1}\right)f(\bar{x}_i)\left[\frac{\partial}{\partial\alpha}\alpha\xi_{\bar{x}^*}(\bar{x}_i)\right]$$

$$= -\sum_{i=1}^{k} f^T(\bar{x}_i)\left[-M^{-1}\left(\xi_{j+1}\right)\left[M\left(\xi_{\bar{x}^*}\right)-M\left(\xi_j\right)\right]M^{-1}\left(\xi_{j+1}\right)\right]f(\bar{x}_i)(1-\alpha)\xi_j(\bar{x}_i)$$

$$-\sum_{i=1}^{k} f^T(\bar{x}_i)M^{-1}\left(\xi_{j+1}\right)f(\bar{x}_i)\left[-\xi_j(\bar{x}_i)\right]$$

$$-\sum_{i=1}^{k} f^T(\bar{x}_i)\left[-M^{-1}\left(\xi_{j+1}\right)\left[M\left(\xi_{\bar{x}^*}\right)-M\left(\xi_j\right)\right]M^{-1}\left(\xi_{j+1}\right)\right]f(\bar{x}_i)\alpha\xi_{\bar{x}^*}(\bar{x}_i)$$

$$-\sum_{i=1}^{k} f^T(\bar{x}_i)M^{-1}\left(\xi_{j+1}\right)f(\bar{x}_i)\xi_{\bar{x}^*}(\bar{x}_i)$$

$$= \sum_{i=1}^{k} f^T(\bar{x}_i)\left[M^{-1}\left(\xi_{j+1}\right)\left[M\left(\xi_{\bar{x}^*}\right)-M\left(\xi_j\right)\right]M^{-1}\left(\xi_{j+1}\right)\right]f(\bar{x}_i)(1-\alpha)\xi_j(\bar{x}_i)$$

$$+\sum_{i=1}^{k} f^T(\bar{x}_i)M^{-1}\left(\xi_{j+1}\right)f(\bar{x}_i)\xi_j(\bar{x}_i)$$

$$+f^T\left(\bar{x}^*\right)\left[M^{-1}\left(\xi_{j+1}\right)\left[M\left(\xi_{\bar{x}^*}\right)-M\left(\xi_j\right)\right]M^{-1}\left(\xi_{j+1}\right)\right]f\left(\bar{x}^*\right)\alpha F$$

$$-f^T\left(\bar{x}^*\right)M^{-1}\left(\xi_{j+1}\right)f\left(\bar{x}^*\right)F$$

Term 3:

$$-N\frac{\partial}{\partial\alpha}E_{\xi_{j+1}}\left[\sigma_{\bar{x}}^2\right] = -N\frac{\partial}{\partial\alpha}\sum_{i=1}^{k}\sigma_{\bar{x}_i}^2\xi_{j+1}(\bar{x}_i)$$

$$= -N\frac{\partial}{\partial\alpha}\sum_{i=1}^{k}\sigma_{\bar{x}_i}^2\left[(1-\alpha)\xi_j(\bar{x}_i)+\alpha\xi_{\bar{x}^*}(\bar{x}_i)\right]$$

$$= -N\sum_{i=1}^{k}\sigma_{\bar{x}_i}^2\left[-\xi_j(\bar{x}_i)+\xi_{\bar{x}^*}(\bar{x}_i)\right]$$

$$= N\sum_{i=1}^{k}\sigma_{\bar{x}_i}^2\xi_j(\bar{x}_i)-NF\sigma_{\bar{x}^*}^2.$$

For instantaneous change, we evaluate derivative at $\alpha = 0$. Note that at $\alpha = 0$, $M^{-1}(\xi_{j+1}) = M^{-1}(\xi_j)$, since $\xi_{j+1} = (1-\alpha)\xi_j + \alpha\xi_{\bar{x}^*} = \xi_j$, and

$M(\xi_{\bar{x}^*}) = \sigma_{\bar{x}^*}^{-2} f(\bar{x}^*) f^T(\bar{x}^*)\xi_{\bar{x}^*} = \sigma_{\bar{x}^*}^{-2} f(\bar{x}^*) f^T(\bar{x}^*) F$.

Combining three terms and evaluating at $\alpha = 0$, we have

$$\frac{\partial}{\partial\alpha} V(\xi_{j+1})\bigg|_{\alpha=0}$$

$$= TR\left[\left\{-M^{-1}(\xi_j)\left[\sigma_{\bar{x}^*}^{-2} f(\bar{x}^*) f^T(\bar{x}^*) F - M(\xi_j)\right] M^{-1}(\xi_j)\right\} W_{\xi_0}\right]$$

$$+\sum_{i=1}^{k} f^T(\bar{x}_i) M^{-1}(\xi_j)\left[\sigma_{\bar{x}^*}^{-2} f(\bar{x}^*) f^T(\bar{x}^*) F - M(\xi_j)\right] M^{-1}(\xi_j) f(\bar{x}_i)\xi_j(\bar{x}_i)$$

$$+TR\left[M^{-1}(\xi_j) W_{\xi_j}\right] - f^T(\bar{x}^*) M^{-1}(\xi_j) f(\bar{x}^*) F$$

$$+NE_{\xi_j}\sigma_{\bar{x}}^2 - NF\sigma_{\bar{x}^*}^2.$$

$$= -TR\left[M^{-1}(\xi_j)\sigma_{\bar{x}^*}^{-2} f(\bar{x}^*) f^T(\bar{x}^*) F M^{-1}(\xi_j) W_{\xi_0}\right]$$

$$+TR\left[M^{-1}(\xi_j) M(\xi_j) M^{-1}(\xi_j) W_{\xi_0}\right]$$

$$+TR\left[M^{-1}(\xi_j)\sigma_{\bar{x}^*}^{-2} f(\bar{x}^*) f^T(\bar{x}^*) F M^{-1}(\xi_j) W_{\xi_j}\right]$$

$$-TR\left[M^{-1}(\xi_j) M(\xi_j) M^{-1}(\xi_j) W_{\xi_j}\right]$$

$$+TR\left[M^{-1}(\xi_j) W_{\xi_j}\right] - f^T(\bar{x}^*) M^{-1}(\xi_j) f(\bar{x}^*) F + NE_{\xi_j}\left[\sigma_{\bar{x}}^2\right] - NF\sigma_{\bar{x}^*}^2.$$

$$= TR\left[M^{-1}(\xi_j) W_{\xi_0}\right]$$

$$+TR\left[M^{-1}(\xi_j)\sigma_{\bar{x}^*}^{-2} f(\bar{x}^*) f^T(\bar{x}^*) F M^{-1}(\xi_j)\left[W_{\xi_j} - W_{\xi_0}\right]\right]$$

$$-F\left[N\sigma_{\bar{x}^*}^2 + f^T(\bar{x}^*) M^{-1}(\xi_j) f(\bar{x}^*)\right] + NE_{\xi_j}\left[\sigma_{\bar{x}}^2\right]$$

$$= TR\left[M^{-1}(\xi_j) W_{\xi_0}\right]$$

$$+TR\left[M^{-1}(\xi_j)f(\bar{x}^*)f^T(\bar{x}^*)F\sigma_{\bar{x}^*}^{-2}M^{-1}(\xi_j)\left[W_{\xi_j}-W_{\xi_0}\right]\right]$$

$$-F\left[N\sigma_{\bar{x}^*}^2+f^T(\bar{x}^*)M^{-1}(\xi_j)f(\bar{x}^*)\right]+NE_{\xi_j}\left[\sigma_{\bar{x}}^2\right]$$

$$=TR\left[M^{-1}(\xi_j)W_{\xi_0}\right]-f^T(\bar{x}^*)\left[F\sigma_{\bar{x}^*}^{-2}M^{-1}(\xi_j)\left[W_{\xi_0}-W_{\xi_j}\right]M^{-1}(\xi_j)\right]f(\bar{x}^*)$$

$$-F\left[N\sigma_{\bar{x}^*}^2+f^T(\bar{x}^*)M^{-1}(\xi_j)f(\bar{x}^*)\right]+NE_{\xi_j}\left[\sigma_{\bar{x}}^2\right]$$

$$=TR\left[M^{-1}(\xi_j)W_{\xi_0}\right]-F\sigma_{\bar{x}^*}^{-2}\left[f^T(\bar{x}^*)M^{-1}(\xi_j)\left[W_{\xi_0}-W_{\xi_j}\right]M^{-1}(\xi_j)\right]f(\bar{x}^*)$$

$$-F\left[N\sigma_{\bar{x}^*}^2+f^T(\bar{x}^*)M^{-1}(\xi_j)f(\bar{x}^*)\right]+NE_{\xi_j}\left[\sigma_{\bar{x}}^2\right]$$

Letting

$$\phi(x,\xi,W)=f^T(\bar{x})\left[M^{-1}(\xi)WM^{-1}(\xi)\right]f(\bar{x})$$

and

$$d(x,\xi)=f^T(\bar{x})M^{-1}(\xi)f(\bar{x}),$$

the result follows.

Steepest descent is obtained by choosing $\bar{x}^*$ to maximize

$$\gamma(\bar{x}^*)=\sigma_{\bar{x}^*}^{-2}\phi\left(\bar{x}^*,\xi_j,W_{\xi_0}-W_{\xi_j}\right)+N\sigma_{\bar{x}^*}^2+d\left(\bar{x}^*,\xi_j^{\cdot}\right).$$

REFERENCES

Bellhouse, D. R. (1987). Model-Based Estimation in Finite Population Sampling. *The American Statistician, 41*, 260-262.

Benson, P. G. and Barry, C. B. (1982). On the Implications of Specification Uncertainty in Forecasting. *Decision Sciences, 13*, 176-184.

Benson, P.G., Pisharody, V. and Yeldan, D. (1985). A Survey of Devices Available for Collecting Traffic Data. Prepared for the Minnesota Department of Transportation.

Benson, P.G., Pisharody, V. and Yeldan, D. (1986). *An Integrated Traffic Data Collection and Analysis System.* St. Paul, MN: Minnesota Department of Transportation.

Blackwell, D. and Girschick, M. A. (1954). *Theory of Games and Statistical Decisions.* New York: Wiley.

Bowley, A. L. (1906). Address to the Economic Science and Statistics Section of the British Association for the Advancement of Science. *Journal of the Royal Statistical Society, 69*, 548-557.

Brewer, K. R. W. (1979). A Class of Robust Sampling Designs for Large-Scale Surveys. *Journal of the American Statistical Association, 74*, 911-915.

Calinski, T. and Harabasz, J. (1974). A Dendrite Method for Cluster Analysis. *Communications in Statistics, 3*, 1-27.

Cassell, C. M., Särndal, C. E. and Wretman, J. H. (1977). *Foundations of Inference in Survey Sampling*. New York: Wiley.

Cochran, W. G. (1977). *Sampling Techniques.* New York: Wiley.

Cook, R. D. and Nachtsheim, C. J. (1980). A Comparison of Algorithms for Constructing Exact D-Optimal Designs. *Technometrics, 22*, 315-324.

Cook, R. D. and Nachtsheim, C. J. (1982). Model Robust, Linear-Optimal Designs. *Technometrics, 24*, 49-54.

Cook, R. D. and Weisberg, S. (1983). Diagnostics for Heteroscedasticity in Regression. *Biometrika, 70*, 1-10.

Davidian, M. and Carroll, R. J. (1987). Variance Function Estimation. *Journal of the American Statistical Association, 82*, 1079-1091.

Evans, J. W. (1979). Computer Augmentation of Experimental Designs to Maximize X'X. *Technometrics, 21*, 321-330.

Fedorov, V. V. (1972). *Theory of Optimal Experiments* . Translated and edited by W.J. Studden and E.M. Klimko, New York: Academic press.

Smith, G. F., Benson, P. G., and Curley, S. P. (1991). Belief, Knowledge, and Uncertainty: A Cognitive Perspective on Subjective Probability. *Organizational Behavior & Human Decision Processes, 8* , 291-321.

Smith, T. M. F. (1976). The Foundations of Survey Sampling: a Review. *Journal of the Royal Statistical Society, Series A, 139*, 183-195.

Tallis, G. M. (1986). On the Optimality of Balanced Sampling. *Statistics & Probability Letters, 4*, 141-144.

Thibodeau, L. A. (1977). *Robust Design for Regression Problems*, unpublished Ph.D. dissertation, University of Minnesota, Dept. of Applied Statistics.

Welch, W. J. (1984). Computer-aided Design of Experiments for Response Estimation. *Technometrics, 26*, 217-224.

Wright, R. L. (1983). Finite Population Sampling with Multivariate Auxiliary Information. *Journal of the American Statistical Association, 78*, 879-884.

Wynn, H. P. (1977a). Minimax Purposive Survey Sampling Design. *Journal of the American Statistical Association, 72*, 655-657.

Wynn, H. P. (1977b). Optimum Designs for Finite Populations Sampling. In S. Gupta and D. Moore (Ed.), *Statistical Decision Theory and Related Topics* (pp. 471-478). New York: Academic Press.