UNIVERSITY OF MINNESOTA

This is to certify that I have examined this copy of a master's thesis by

Ning Li

And have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

____

Name of Faculty Advisor

____

Signature of Faculty Advisor

____

Date

GRADUATE SCHOOL

A FREIGHT DEMAND MODEL FOR RURAL MINNESOTA

A THESIS
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

NING LI

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

Professor David M. Levinson, Advisor

May, 2005

ACKNOWLEDGEMENTS

I would like to acknowledge my advisor, Dr. David Levinson for providing me the great opportunity of working on the study of the Cost/Benefit Study of Spring Load Restriction of which this freight demand model was part. I would also like to thank to the Minnesota Local Road Research Board (LRRB) and the Minnesota Department of Transportation (Mn/DOT) for their financial support of this study. The support and guidance of the fellow graduate students who have worked on the Cost/Benefit study is also greatly appreciated.

ABSTRACT

Spring Load Restrictions (SLR) impose load restrictions on heavy trucks during the spring thaw period. Although the policies have been implemented for many years, their economic effect on truckers has been unclear. Mn/DOT and the LRRB initiated a Benefit/Cost Analysis of Spring Load Restrictions, for which a freight demand model (FDM) is required, to measure the effects of the SLR policy quantitatively. In this thesis, a freight demand model is built to estimate the impacts of SLR on freight transportation. The FDM adopts the traditional four-step transportation planning techniques and utilizes transportation software, EMME/2, to conduct route assignment. The model allows various policy scenarios to be tested before being tested in practice. The model shows the SLR policy increased truck Vehicle Kilometers of Travel (VKT) in Lyon County, Minnesota by about 30.4% percent.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

Spring Load Restrictions (SLR) restrict the axle loading of heavy trucks during the spring thaw. SLR (under various names, including spring (seasonal) weight restrictions, spring bands, or spring thaw load restrictions) are implemented in many cold climate countries, including the United States, Canada, France, Norway, Finland, and Sweden. The policy aims to minimize pavement damage and extend the useful life of roads, which enables road authorities to save on infrastructure investment and maintenance of roads.

In cold regions, the strength of pavement varies seasonally. During the winter, the soil under the pavement surface hardens when the temperature drops below the freezing point. When spring arrives, the frozen soil thaws and is in a saturated condition. Under this condition, the soil under the pavement becomes weak and reduces the bearing capacity of the pavement. (Yukon Infrastructure, 2002) At this time, heavy loads beyond the bearing capacity will cause excessive pavement damage and reduce road life.

To solve this problem, perhaps the most obvious technical solution is to improve pavement capacity of all roads so that they can bear heavy loads even during the spring thaw period. But this is expensive for the responsible agency, and therefore SLR policies are implemented to reduce pavement damage caused by heavy trucks during the thawing period.

Although SLR can reduce pavement damage caused by heavy vehicles, the SLR policy also imposes costs on the trucking industry due to detouring or increased number of truckloads. A question naturally arises: does the benefit really exceed the cost? Although the SLR policy has been implemented for many years, we are still unclear on this issue. The trucking industry complains the SLR policy imposes costs and inconvenience while the road agency strongly advocates this policy because they believe it reduces required pavement investment and maintenance significantly.

Quantifying the cost and benefit of SLR is difficult because many costs and benefits are intangible and cannot be directly measured. Previous studies gave controversial conclusions on this issue. Mn/DOT initiated a research project to conduct Benefit/Cost Analysis of Spring Load Restrictions Policy. The main objective is to quantitatively evaluate the effects of SLR in Minnesota and thus find the solution with the minimum total social cost to users and the road agency. A key task of the analysis is to build a freight demand model, which can measure the effects of SLR policy quantitatively.

In this thesis, a freight demand model was developed to evaluate the effects of the Spring Load Restrictions policy in Minnesota. Lyon County, Minnesota was used as an example to illustrate the model. A survey was conducted to provide background information and some parameters for modeling beforehand. The traditional four-step model techniques, which include trip generation, trip distribution, vehicle type choice, and route assignment are implemented here. EMME/2 transportation planning software is used to conduct the trip assignment procedure.

CHAPTER 2: THE ROLE OF FREIGHT DEMAND MODEL

Lyon County, Minnesota is modeled first to test our methodology and is illustrated in detail in this thesis. Clay and Olmsted counties in Minnesota are also modeled to estimate the economic effect of SLR policy in Minnesota. Those counties represent typical regions of rural Minnesota.

A flowchart of the framework for analyzing the Benefit/Cost of SLR is shown in Figure 1 and is detailed below.

The first step is to obtain the data needed for modeling. A Lyon County GIS map with traffic volume on most of the roads is obtained from the County Engineer, together with a detailed road restriction map.(Public Works of Lyon County, 2003) The GIS map is transformed to EMME/2 format by virtue of Arc/Info and Matlab programming. (Appendix 1) Freight facilities in Lyon County are located in the map through the Mn/DOT Freight Facilities Database.

A two-round survey was conducted in the year 2003 using both mail and on-site interview methods.(Smalkoski, 2003) The object of the survey is to provide background information, parameters like truck operating cost, value of time, and trips generated for each freight facility type, which could be used in the Benefit/Cost analysis.

A freight demand model is implemented to emulate the truck freight pattern in Lyon County. The model calculates the truck trip demand generated in each freight facility within the county, determines their destinations and vehicle chosen based on summary data derived from the survey, and assigns them on each link. The truck volume on each link can then be obtained under two scenarios: without SLR and with SLR.

The total truck Vehicle Kilometers of Travel can be calculated for each scenario, and the increased truck VKT due to SLR implementation can be converted to costs using the parameter of truck operating cost (Hashami, 2004) derived from the survey.

A pavement team worked on the Pavement Performance Model simultaneously. The result of the freight demand model, together with other data like pavement material, moisture, and temperature are used as input for the economic model. The model will estimate the pavement life under the two scenarios and the economic benefits of pavement life extension can be derived.

Knowing these cost and benefits, the economic impacts of the SLR policy can be estimated.

Flowchart of SLR Benefit/Cost Analysis

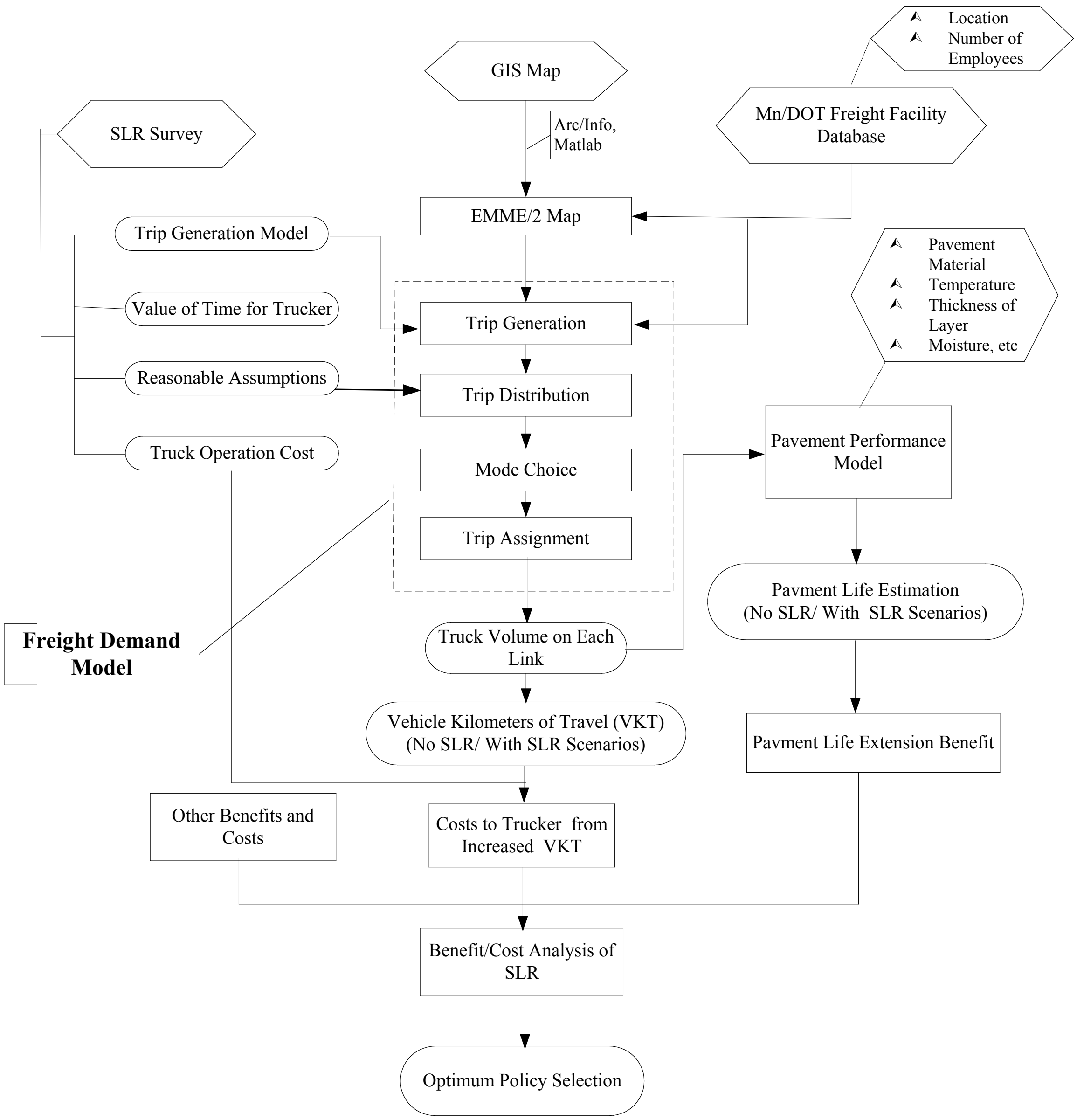

FIGURE 1 Flow chart of the SLR Benefit/Cost Analysis

CHAPTER 3: FREIGHT DEMAND MODEL

3.1 Introduction

To estimate the impact of SLR quantitatively, it is necessary to build a freight demand model to measure how truck Vehicle Kilometers of Travel (VKT) changes during the SLR period. Lyon, Olmsted and Clay Counties are modeled to represent typical areas of Minnesota.

Lyon County, Minnesota is modeled first to test the methodology and it is used as an example in this thesis. The freight demand model is detailed below.

The first step is to obtain the data needed for modeling. A GIS map with total traffic volume on most of the roads is obtained from the county GIS coordinator, together with a detailed road restriction map (Public Works of Lyon County, 2000, 2003). The GIS map is transformed to EMME/2 format using Arc/Info and Matlab programming (Li, Zou, Levinson, 2004). Freight facilities in the county are located using the Mn/DOT Freight Facilities Database.

A two-round survey was conducted in 2003 using both mail-out/mail-back and on-site interview methods. (Smalkoski, 2003) The object of the survey was to provide background information, parameters like truck total operating cost, value of time, and truck trip generation rate for each freight facility type, which could be used in the Benefit/Cost analysis.

The freight demand model is implemented to emulate the truck freight patterns in three Minnesota counties (Lyon, Olmsted, Clay). The model calculates truck trip demands generated in each freight facility within each county, determines their destinations and truck type used based on data derived from the survey, and assigns them to each link.

A transportation-planning software package, EMME/2, is used to assign the freight demand model's traffic to the network. The roads in each county are classified into 4 types during the SLR period: 5-ton roads, 7-ton roads, 9-ton roads and 10-ton roads. Outside the SLR period, most 5, 7, and 9-ton roads can accommodate 9-ton axle loadings, (and 10-ton roads can accommodate 10-ton axle loadings). (Minnesota Department of Public Safety, 2003) These numbers represent the maximum allowable axle weight limits during the two periods. In the basic analysis, the freight demand model is run using two scenarios. The first scenario does not have SLR policy (without SLR), indicating that 9-ton trucks can run on all roads without restriction. The second scenario (with SLR) imposes the SLR policy (with 100% compliance) so that the all trucks are subject to the load restriction policy. The two scenarios are compared to see how SLR changes truck traffic patterns and VKT.

The total truck Vehicle Kilometers of Travel can be calculated for each scenario. In the Benefit/Cost analysis, the change in truck travel due to SLR can be converted to costs using the truck operating cost estimated from the survey. (Hashami, 2003) Similarly, the change in truck travel can be converted to changes in pavement damage, using the pavement performance model, and can be considered benefits for road owners associated with the existing SLR policy.

3.2. Truck fleet composition

The freight demand model and pavement performance model require vehicle classification counts and the truck fleet composition. To simplify the modeling process, three types of trucks are defined: 2-axle, 3-axle, and 5-axle. Truck parameters are listed in Table 1. It is worth noting that the gross weight and tare weight data comes from a Canadian study (TRIMAC Consulting Service, 1999) and weight distribution per axle data comes from the Iowa Department of Transportation (Iowa DOT, 1985).

Table 1 – Truck parameters in the model

Truck Type	Gross Weight (ton)	Tare Weight (ton)	Front Axle Type	Middle Axle Type	Rear Axle Type	Weight Distribution per axles (empty truck)	Weight Distribution per axles (full truck)
2 axles	12	3.4	Steer	-	Dual wheels	1.00:1.15	1.00:1.65
3 axles	21	8.0	Steer	-	Dual Tandem	1.00:1:36	1.00:1.98
5 axles	39	14.4	Steer	Dual Tandem	Dual Tandem	1.00:1.50:1.00	1.00:3.09:3.07

Detailed truck classification and traffic counts were conducted by Mn/DOT both during and after the 2004 SLR period in Lyon County. Counts collected during the SLR period in Lyon County are used to calculate the truck fleet composition, which shows the percent of each category of truck in the fleet. Counts were taken at 63 sites throughout Lyon County for 48 hours at each site. The counts were halved to give the 24-hour ADT for each site. The sample consisted of ten 5-ton roads, twelve 7-ton roads, seven 9-ton roads, and twenty-one 10-ton roads. Counts were conducted at another ten sites, but these were considered unusable by Mn/DOT. Another three sites were also not reported. Of the sites not used, two were 10-ton roads, four were 9-ton roads, two were 7-ton roads, and five were 5-ton roads. Three sites reported no trucks over the 48-hour time and these sites were included in the average truck ADT, but were excluded from the truck category breakdown.

For each site, data was available for passenger vehicles per day, single unit 2-axle trucks, single unit 3+-axle trucks, 3-axle semis, 4-axle semis, 5+-axle semis, trucks with trailers and buses, and trucks with twin trailers.

As noted above, the trucks were categorized into 2-axle, 3-axle, and 5+-axle. All 2-axle single unit trucks, trucks with trailers, and buses were categorized as 2-axle vehicles. All 3-axle single unit trucks and 3-axle semis were categorized as 3-axle vehicles. The 4-axle semis were made up a very small percentage of all truck traffic, and were categorized as 3-axle vehicles. All 5+-axle semis were combined with twin trailer traffic to obtain the category for 5-axle vehicles. The percentages of each of the three categories of trucks for each of the categories of roads as well as the truck ADT for each category are in Table 2. This truck fleet composition is adopted in the freight demand model.

According to the above truck parameter and truck fleet composition, the actual carrying capacity of each type of truck on each different road type can be calculated.

Table 2 – Truck ADT and breakdown of trucks by road category

Road Category	Truck ADT	% Trucks	% 2-axle	% 3-axle	% 5-axle
5-ton	4	6.5%	85.4%	14.6%	0%
7-ton	29	8.3%	54.2%	29.7%	16.1%
9-ton	27	9.1%	47.5%	27.1%	25.3%
10-ton	410	17.3%	24.8%	9.8%	65.4%

In Table 1, truck net vehicle weight, gross vehicle weight and payload parameters come from a Canadian study by TRIMAC Consulting Service (1999). The weight restriction is calculated using the "Restricted Gross Weight Table" from Mn/DOT (2004). Axle weight distribution data is from the 1985 Truck Weight Index from the Iowa Department of Transportation as shown in Table 1 (Iowa DOT, 1985). The net weight that a truck can carry is the minimum of the gross weight limit and gross vehicle weight minus vehicle tare weight. Truck fleet composition used in subsequent modeling comes from Table 3.

3.3. Road network and zone structure

To run the model, it is necessary to define a network and traffic zone structure. The network geometry needs to be coded to a standard node-link format, such as that used by EMME/2. The easiest way to do that is to transform a network obtained from GIS maps into this node-link format, using a program developed in Matlab, and described in Appendices 1 and 2 (Li, Zou, Levinson, 2004). In Lyon County, there are 225 traffic analysis zones (TAZs) evenly located within the county. Since Lyon County is largely an agricultural county, we think of each of these TAZs as a *virtual farm*. In Olmsted County there are 212 TAZs, and in Clay County there are 222 TAZs. In Clay County, the traffic analysis zones are evenly spaced among a 16 x 14 matrix. Each zone is approximately

3256 m (x-direction) by 3502 m (y direction). Two of the zones that were not close

Table 3 - Truck fleet parameter on each type of road

Truck Configuration	Net Vehicle Weight (ton)	Gross Vehicle Weight (ton)	Payload (ton)	Weight Restriction (ton)	Actual carrying capacity (ton)	Proportion of fleet
5 ton road	**Carrying capacity of a typical 5 ton truck**				**4.0**	
2 Axle	3.4	12	8.6	7.3	3.9	85.4%
3 Axle	8	21	13	12.9	4.9	14.6%
7 ton route	**Carrying capacity of a typical 7 ton truck**				**8.8**	
2 Axle	3.4	12	8.6	10.2	6.8	54.2%
3 Axle	8	21	13	18.1	10.1	29.7%
5 Axle	14.4	39	24.6	27.8	13.4	16.1%
9 ton route	**Carrying capacity of a typical 9 ton truck**				**13.0**	
2 Axle	3.4	12	8.6	13.1	8.6	47.5%
3 Axle	8	21	13	23.6	13	27.1%
5 Axle	14.4	39	24.6	35.7	21.3	25.3%
10 ton route	**Carrying capacity of a typical 10 ton truck**				**17.4**	
2 Axle	3.4	12	8.6	>13.1	8.6	24.8%
3 Axle	8	21	13	>23.6	13	9.8%
5 Axle	14.4	39	24.6	35.7	24.6	65.4%

enough to any roads (the north-central portion of the map) were deleted to make room for more external stations. Network summary statistics are given in Table 4.

Table 4 – Descriptive Network Statistics

County	Lyon	Olmsted	Clay
Total number of zones	250	250	250
Internal zones	225	212	222
External zones	25	38	28
Number of Freight facilities	59	126	63
Number of Links	4494	5032	5888
(including centroid connectors)			
Number of Regular Nodes	1379	1866	1891
10 ton roads km (mile)	266 (166)	309 (193)	332 (208)
9 ton roads km (mile)	191 (119)	183 (114)	241 (151)
7 ton roads km (mile)	260 (163)	604 (378)	123 (77)
5 ton roads km (mile)	1437 (898)	818 (511)	2286 (1429)

The freight facilities in Lyon County were located using the Mn/DOT freight facility database and input into the transportation network. In Lyon and Olmsted counties, each freight facility is assigned to a unique Traffic Analysis Zone (TAZ). In Clay County, each of the firms generating trips is assigned to the TAZ with the nearest centroid. When connecting centroids to the network, it is assumed that any and all freight facilities and trips generated within a TAZ have access to the highest rated (least restricted) road available in that zone. Thus, for instance, if a TAZ has both 5-ton and 9-ton roads available, the centroid will be connected to the 9-ton road.

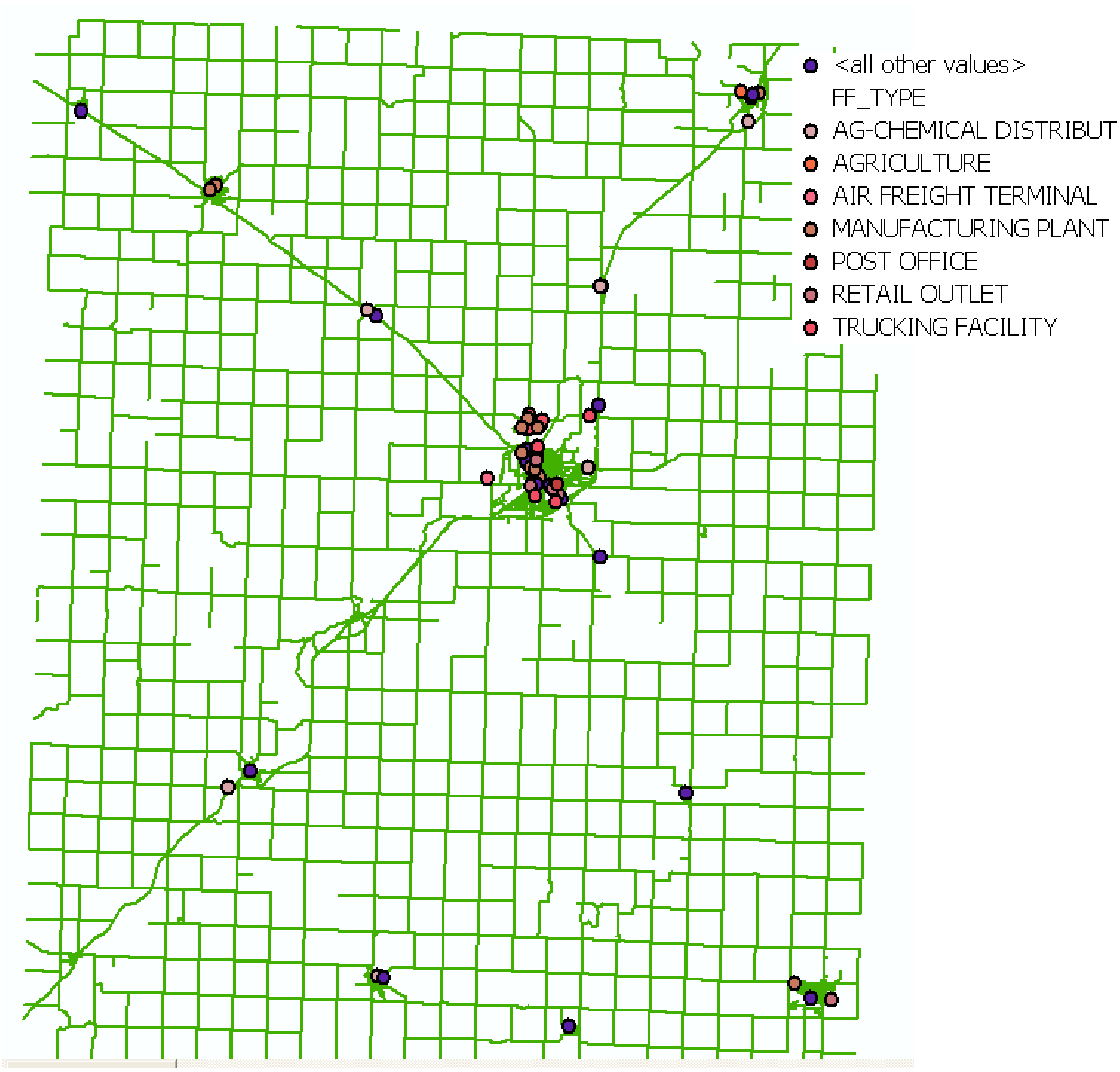

Figure 2 Freight facility locations in Lyon County

In the network, four corresponding types of modes: ‘c’,’l’,’m’,’h’ were set up. Mode ‘c’ represents trucks with a small load (5-tons per axle or less), which can run on all four types of roads. Mode “l” represents trucks with light loads (7-tons per axle or less), which can run on 7-ton or higher-level roads. Mode ‘m’ represents trucks with moderate loads (9-tons per axle or less), which can run on 9-ton or higher-level roads. Mode ‘h’ represents heavy trucks, which can only run on 10-ton roads.

In accordance with the SLR period road restriction map (Public Works of Lyon County, 2000, 2003), different modes are assigned to each link in the road network as link attributes, resulting in a road network with a hierarchy of 5, 7, 9 and 10-ton roads.

A different speed function is assigned to the different level of roads. It is worth noting here that the speed specified on each type of road is only used to determine the fastest route in the trip assignment process. Thus, the absolute value of speed does not much affect the result of the freight demand model. Only the relative speed difference affects the results of the freight demand model. Generally speaking, higher grade roads have higher average speed than lower grade roads. For instance, many 9-ton roads have shoulders, which give drivers comfort while traveling at a higher speed. Here 5, 7, 9 and 10-ton roads are assumed to have speeds of 48, 64, 80, and 96 km/h (30, 40, 50 and 60 mph), respectively. The speed specified here is just used for modeling of truck demand (i.e. the flows were calibrated, and the speeds were adjusted to make the flows match better) and is not necessarily the actual running speed on these roads.

3.4. Trip Generation

Trip generation requires estimating the truck demand generated within the county. It is assumed that the truck demand is generated from various kinds of freight facilities within the county. According to the Mn/DOT Freight Facility Database classification, there are 8 land use categories associated with freight transportation: Farm, Agriculture Chemical Center, Grain Elevator, Manufacturing Plant, Retail Outlet, Trucking Facility, Wholesale Distribution Center, and Other Freight Facilities. Figures 2 maps the locations of these facilities in Lyon County.

3.4.1 Trip generation for non-farm facilities

In the process of developing a thorough freight demand model for Minnesota as part of the Spring Load Restriction (SLR) Cost-Benefit Study, research was undertaken to develop freight generation models by freight facility type. This section will provide an overview of previous freight generation studies, the methodology used in the surveying and analysis process, and the results (Smalkoski, 2003).

Mirjam *et al.* (2002) introduce a freight trip generation model from their native Netherlands. They noted that freight trip generation has been given little research over the past though it accounts for a growing percentage of the traffic congestion on today's roads. They discovered that in different industry sectors trip intensities can vary by an order of magnitude, thus a trip generation model should be fit for each sector independently. They used employee count and site area as independent variables describing trip generation depending upon the type of activity performed at the firm. The model of choice was simple linear regression. Simple linear regression is also used in ITE's Trip Generation Manual (2003), where the independent variable used to describe the number of trips varies by facility.

As noted before, a survey was mailed to 2,523 freight industry candidates during the first half of 2003. (Smalkoski, 2003) The thirteenth question on the long form asks, "How many truck loads did your firm carry last year?" The goal of the analysis was to explain the variance in this number as a function of a scale variable. The fourth question asks, "How many direct employees does your firm have?" The number of direct employees was chosen as the scale variable because the data provided by the firms is thought to be more precise than other responses, and it has been shown to explain trip rates in previous studies.

The data was further divided into seven common freight facility types by matching the firm with its listing in the freight facilities database and analysis of the products that the firms' trucks carry. The seven types are: Agriculture Chemical Distribution Center, Agriculture, Grain Elevator, Manufacturing, Trucking Facility, Waste and Recycling Center, Wholesale Distribution Center. Table 5 displays descriptive statistics for the responses received from these freight facility types.

Three model functional forms were tested in this analysis. Simple linear regression has been used in all previous studies. Cobb-Douglas models and quadratic models account for some non-linearity; these were also tested to see if the fit to the data improved. The number of truckloads (*TL*) carried was the dependent variable and the number of direct employees (*DE*) on staff was the independent variable. All nil and zero responses were eliminated prior to model fitting.

Linear Regression: $TL = \beta_1 + \beta_2 * DE$

Quadratic Model: $TL = \beta_1 + \beta_2 * DE + \beta_3 * DE^2$

Cobb-Douglas Model: $TL = e^{\beta 1} \times DE^{\beta 2}$

Table 5 – Descriptive Freight Facility Statistics

		Direct Employees		Truck Loads	
Facility Type	N	Mean	Standard Deviation	Mean	Standard Deviation
Ag Chem	18	23	32	1866	1480
Agriculture	28	28	52	2342	3282
Grain	38	20	25	3410	4700
Manufacturing	15	569	1925	5005	8602
Trucking	90	63	134	16471	33619
Waste	4	68	38	25280	27891
Wholesale	13	216	289	18367	41536

After fitting three options, shown in Table 6, it was determined that the Cobb-Douglas model provides the best fit. The Cobb-Douglas model accounts for non-linearity in the relationship between the number of employees and the number of truckloads. The relationship varies greatly from freight facility to freight facility, thus it is best to use a separate model for each facility type.

The model fit well with the exception of the manufacturing freight facility type. This is the best model that is available for these data, however in further research this freight facility type should be further broken down to account for the wide variation among firms.

Table 6 – Trip Generation Models

Industry Type	N	ß1	t-statistic	ß 2	t-statistic	ß 3	t-statistic	R^2
				Linear Regression				
Ag Chem	18	1355.90	3.43	22.18	2.16	-	-	0.225
Agriculture	28	2073.22	2.90	9.58	0.78	-	-	0.023
Grain	38	946.75	1.29	123.31	5.38	-	-	0.446
Manufacturing	15	4629.34	1.94	0.66	0.54	-	-	0.022
Trucking	90	5333.62	1.91	176.23	9.31	-	-	0.496
Waste	4	-23850.53	-3.54	727.86	8.11	-	-	0.971
Wholesale	13	-1358.44	-0.12	91.22	2.73	-	-	0.404
				Quadratic Model				
Ag Chem	18	711.03	1.51	82.64	2.74	-0.51	-2.11	0.402
Agriculture	28	764.06	0.93	132.69	2.70	-0.64	-2.58	0.228
Grain	38	-22.11	-0.02	245.60	2.64	-1.43	-1.35	0.474
Manufacturing	15	5218.77	1.96	-8.22	-0.52	0.00	0.56	0.047
Trucking	90	-749.01	-0.25	388.83	7.08	-0.34	-4.08	0.577
Waste	4	-2120.00	-0.23	38.33	0.14	4.41	2.55	0.996
Wholesale	13	-7447.02	-0.57	198.82	1.67	-0.14	-0.94	0.453
				Cobb-Douglas Model				
Ag Chem	18	5.41	10.82	0.68	3.70	-	-	0.461
Agriculture	28	5.05	9.83	0.77	4.01	-	-	0.382
Grain	38	5.47	12.76	0.78	4.75	-	-	0.385
Manufacturing	15	5.35	3.30	0.26	0.66	-	-	0.033
Trucking	90	5.67	23.71	0.92	12.73	-	-	0.648
Waste	4	0.48	0.72	2.23	13.67	-	-	0.989
Wholesale	13	1.87	2.28	1.34	7.50	-	-	0.836

For retail facilities, which are not in the model above, a linear trip generation model described in Mirjam's report was used (Mirjam, 2002), which gives detailed freight trip generation rates classified by firm types. It is assumed these retail facilities are food stores.

Table 7 shows summary results under the without SLR scenario. Under the SLR scenario, truckers have to increase truck trips if they choose the lower level roads. The truck trip increase is determined by the actual carrying capacity of each type of roads. For instance, assuming the above proportion, the average carrying capacity on a 7-ton road is 8.8 tons per truck while on a 9-ton road the capacity is 13.0 tons per truck. Theoretically, during SLR, if only a 7-ton route is available, the trucker has to undertake 1.48 times the normal number of truck trips (1.48=13/8.8). Similarly, the trucker has to use 3.25 times as many truck trips if a 5-ton route is chosen (3.25=13/4.0). This overestimates the truck trip during the SLR period for the following reasons:

Table 7 – Summary Truck trip rate of other freight facilities

Freight Catalog	Minnesota model
Agriculture chemical distribution center	129.84
Grain elevators*	83.4
Manufacture plant	29.0
Retail outlet	128.7
Trucking facility	388.4
Wholesale outlet	29.87

**Note that we do not adopt the grain elevator truck trip rate from the trip generation model. According to our survey, many grain elevators use rail to transport grain. Through talking to staff in several grain elevators, 60% of the outgoing freight from grain elevators are assumed to use trucks and the remaining 40% rail. The trips required from a grain elevator are based on the freight entering that elevator, which is considered more accurate than using a trip rate based on number of employees. The amount of grain leaving by truck is the amount of grain entering discounted by 40%. The number of trucks required is computed based on tonnage.*

- Some industries will choose to shift cargo transportation to the No SLR period. They do not transport cargo during the period. For example, some farms will store grain during SLR period.

- The SLR does not affect some industries because their products, when fully loaded onto trucks, do not exceed the load limits.

The SLR survey provided some information on how each industry is affected by SLR as shown in Table 8. The table lists the percentages of industries that are affected by SLR and must reduce their load size. The truckload increase factors due to SLR are calculated as follows. It is assumed only a percentage of trucks are affected by SLR for each industry. Among those affected, only a certain proportion reduced load size, the other trucks shifted their timing to avoid SLR.

The reported industry type from the survey was categorized to be consistent with the freight facility category in the freight demand model. From this the average increase factor in each industry was calculated. For industries that are not included in the survey, the sample average was used to approximate the increase factor.

Although waste and recycling centers and post offices are missing from the freight facility database, their existence is still assumed. It is assumed a 2-axle postal truck will run on all roads once a day in both with and without SLR scenarios, and a 3-axle garbage truck will cover all roads once a week in the without SLR scenario and twice a week during the SLR period.

Table 8 – SLR impacts on industry

Industry	Count	Affected by SLR	Reduce Load Size	Industry reclassification	7-ton road increase factor	5-ton road increase factor
Agricultural	99	86.9%	79.8%	Agriculture	1.16	2.38
Agriculture Chem	28	100.0%	96.4%	Chemical	1.43	3.13
Aggregate	16	87.5%	75.0%	Manufacture		
Timber	7	85.7%	85.7%	Manufacture		
Construction	23	82.6%	43.5%	Manufacture	1.01	2.04
Beverages	5	100.0%	40.0%	Retail		
Petroleum	19	89.5%	73.7%	Retail		
Food Products	20	60.0%	40.0%	Retail	0.81	1.57
Dairy	6	66.6%	33.3%	Wholesale		
General Products	52	46.2%	19.2%	Wholesale		
Paper	7	28.6%	14.3%	Wholesale		
Industrial Supplies	32	21.9%	18.8%	Wholesale	0.74	0.91
Overall	315	71.4%	79.1%		1.12	2.12

3.4.2 Truck trips generated by farms

Truck trips generated by farms are associated with the amount of agricultural product that needs to be transported. First the total production of grain in Lyon County was obtained, followed by a determination of the truck fleet carrying the grain. From this data, the truck trip rates can be calculated. Crop production for each of the counties (Lyon, Olmsted and Clay) for the year 2001 are summarized in Table 9 (National Agricultural Statistical Service, 2001).

Table 9 – Major Crop Production

Crop	Lyon County (Tonnes)	Olmsted County (Tonnes)	Clay County (Tonnes)
Corn	558,911	391,429	82,973
Soybean	140,593	68,875	106,005
All wheat	4,515	0	221,869
Oats	2,465	5882	819
Total	706,484	466,186	411,666

In Lyon County, assuming the crops are evenly distributed among the 225 virtual farms and are evenly transported each day, 8.6 tonnes needs to be transported from each farm every day. (8.6=706,484/(225*365)).

In Clay County, assuming the crops are evenly distributed among the 222 virtual farms and are evenly transported each day, 5.08 tonnes needs to be transported from each farm every day. (5.08=411,666/ (222*365)).

During the SLR period, farms may transport less grain than usual to avoid extra shipping costs. To explore the fluctuation of grain hauling due to SLR, data was obtained from All-American Coop, which runs grain elevators in Stewartville and Viola, both in

Olmsted County, Minnesota.

A comparison was made between the quantity of commodities hauled to and from the elevator in April and the commodities hauled during the other months of the year. April was specifically chosen because it is the only month that SLR is in effect for the entire month. March and May, the months before and after April did not experience a decrease due to SLR because extra loads could be hauled immediately before and after the SLR period. For commodities hauled from the elevator from 1995 to 2003, there was an 8.5% decrease in hauled-in tons during April compared to the average tons hauled in other months. The monthly fluctuation of tons hauled out is illustrated in Figure 3. The haul-in data also showed a decrease in the month of April. This decrease was more pronounced at 11.5%. Monthly tonnage is illustrated in Figure 4. Because there is a conservation of mass, over time, the tons hauled in equals the tons hauled out. Because of this equality, the average of the two percentages — 10% — approximates the percent decrease of tons hauled during the month of April. This 10% percent decrease of grain hauling during SLR is taken into account in the modeling.

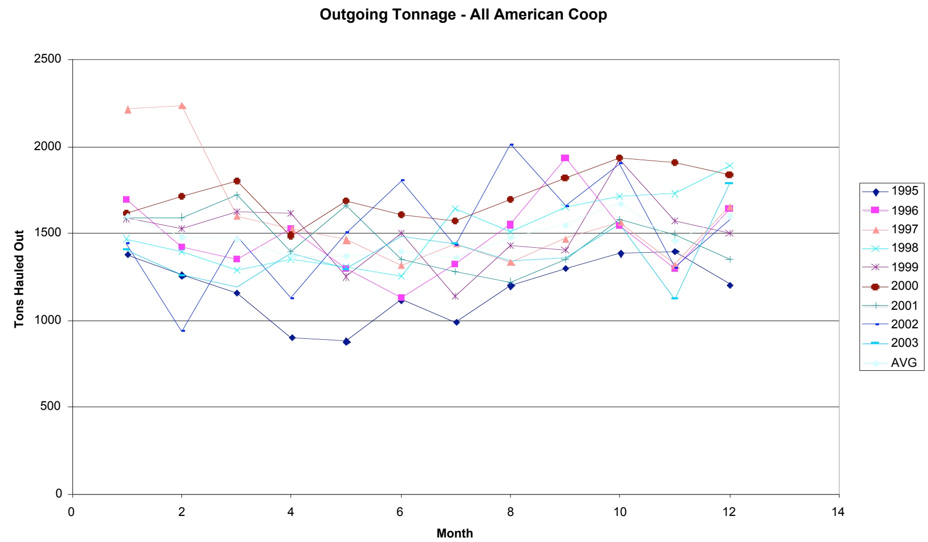

Figure 3: Outgoing tonnage from All American Co-op in Olmsted County 1995-2003.

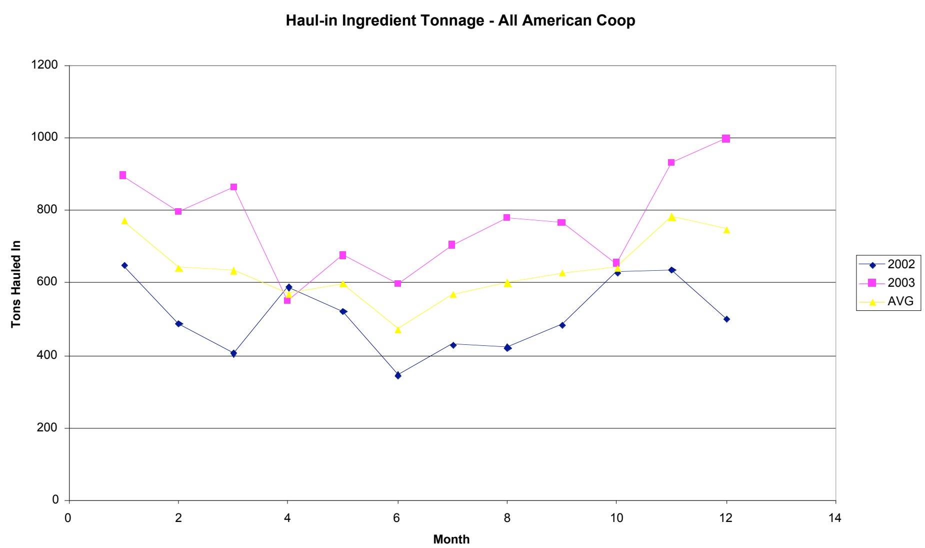

Figure 4: Incoming tonnages from All American Co-op in Olmsted County 2002-2003.

Considering the seasonal fluctuation of grain hauling, in Lyon County, 8.67 tonnes of grain per day will be hauled in the period without SLR and 7.88 tonnes grain per day will be hauled during the SLR period. In Clay County 4.65 tonnes would need to be transported each day during the month of April, while 5.12 tonnes needs to be delivered during the remainder of the year.

If SLR is not implemented, 9-ton routes are available for farms to carry grain to nearest grain elevator. The carrying capacity of a typical 9-ton truck is 13.0 tons as shown in Table 1. The daily truck trip for each farm outside the SLR period is shown in Table 8.

Table 10 – Trip Rates for Farms

Highest Road Accessible in Zone	Lyon County	Olmsted County	Clay County
Outside the Spring Load Restriction Period			
9-ton	0.67=8.67/13.0	0.46=6.02/13.0	0.39=5.08/13.0
During the Spring Load Restriction Period			
5-ton	1.97 = 7.88/4.0	1.37 = 5.47/4.0	1.16 = 4.65/4.0
7-ton	0.90 = 7.88/8.8	0.62 = 5.47/8.8	0.53 = 4.65/8.8
9-ton	0.61 = 7.88/13.0	0.42 = 5.47/13.0	0.36 = 4.65/13.0

If SLR is implemented, each farm has to determine what kind of roads to use to reach its destination. Table 1 shows average carrying capacity of a typical truck on each type of road. In Lyon County 7.88 tonnes of grain need to be carried out for each farm every day during the SLR period. The trip rates during the SLR period are shown in Table 10.

3.5. Trip Distribution

Trip distribution is based on the origin-destination sketch map in Figure 5. It is assumed that each farm will deliver their grain product to the nearest grain elevator and return empty. Each farm receives deliveries from the nearest Agricultural Chemical (Ag-Chem) facility and the trucks return to the Ag-Chem facilities empty. For the other types of freight facilities, it is assumed all traffic is bound for external stations. The trips to each external station will be distributed in proportion to the real traffic count at these external points. There is also external to external traffic, which is assumed to be 20% percent of the total traffic stream at each exit. A Matlab program (Appendix 3) was written to realize the above function.

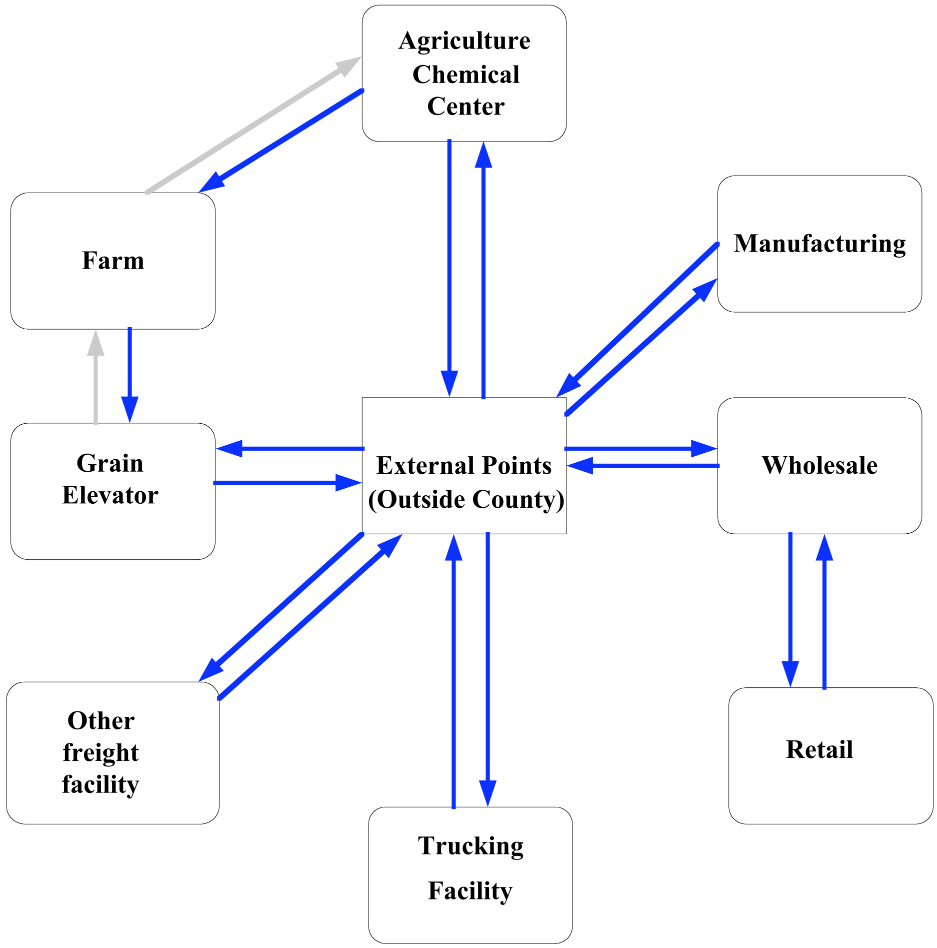

Figure 5 Freight demand pattern in Lyon County

3.6. Vehicle Type Assignment

It is assumed that truckers will choose the most economically beneficial vehicle. In the absence of restrictions, truckers will tend to choose the heaviest trucks they have so they can carry more goods. However, weight restrictions may prevent this, especially if a trucker would face a significant amount of detouring. Truckers faced with weight restrictions must compare the costs of detouring versus the costs of using trucks with a lower weight or payload capacity, which may result in using more trucks. The total cost for a trip is:

$$C = (T_T + T_L) * N * c$$

where:

C – Total cost

T_T – Travel time for each trip (hour)

T_L – Time for loading and unloading (hour)

N – Number of Truckloads

c – Value of time (dollar per hour)

It is assumed based on discussion with shippers that 30 minutes is required for loading and unloading cargo for each truckload. The trucker will find the route that has the least total cost. For example, if a 9-ton route is the highest level of route they can find between the origin and destination, they will first calculate the cost of using 9 ton route, which is named c_1, and then check to see if cost of using other route, a 7-ton route for instance (c_2) will lower their costs. Choosing a 7-ton truck means they will have to use 1.48 = 13/8.8 truckloads instead of 1 and they will spend 18.9 = 0.69*30 minutes more on loading and unloading, but they will travel over a shorter route. The trucking firm will calculate the total cost of each choice (5,7,9 ton route) and then select the most economical one.

3.7. Route Assignment

In the route assignment, it is assumed that truckers will behave according to user equilibrium assignment theory in which they will choose routes with the least travel time (T). Since rural areas are being modeled, congestion effects are ignored, which makes this equivalent to an all-or-nothing shortest path assignment.

$$T = 60 * L / v$$

T: Travel time (minutes).

L: road section length (kilometer)

V: vehicle speed (km/h), assumed to be 48, 64, 80 and 96 km/h (30, 40, 50, and 60 mph) for 5, 7, 9 and 10-ton roads respectively.

The constant "60" converts the unit of hour to minutes.

It should be noted that interstate highways are assigned a speed of 104 km/hr (65 mph), which distinguishes them from other 10 ton roads.

CHAPTER 4: VALIDATATION AND CALIBRATION

The freight demand model is run using the Emme/2 software, providing truck volumes for each road section. It is important to know the accuracy of the model.

Two methods are used to check and calibrate the model. The first method is to use the observed total traffic count map. Lyon County data is shown in Figure 6**.** If trucks are assumed to be a certain proportion of the total traffic spread evenly on the roads, the truck pattern should be similar to the real total traffic pattern. Fortunately, the result of our

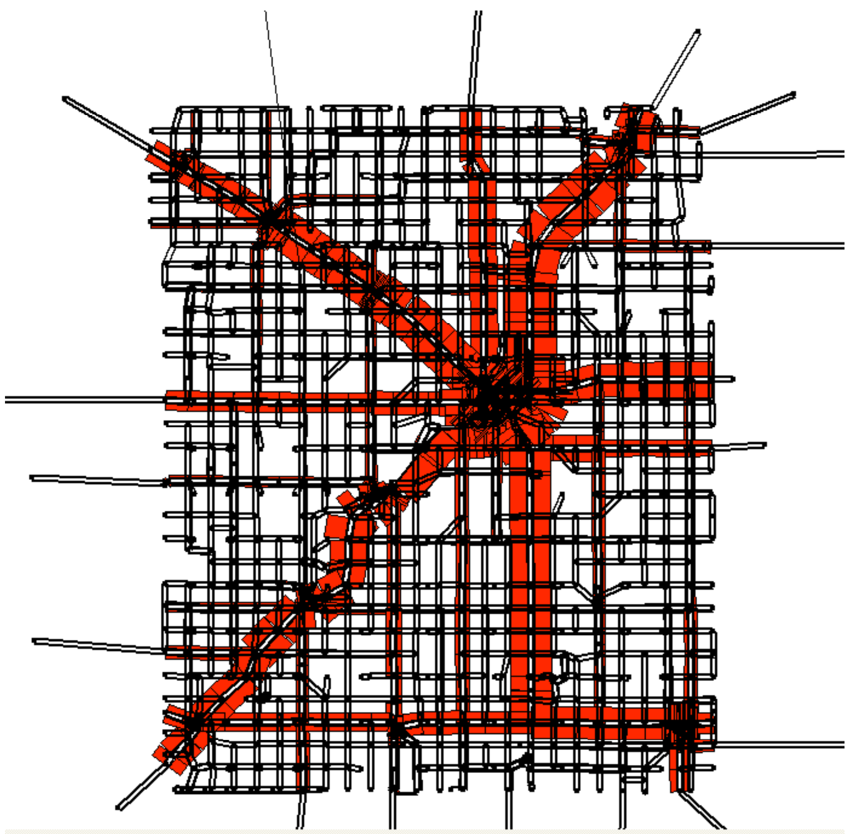

Figure 6: Total observed traffic counts in Lyon County

freight demand model (without SLR scenario) (Figure 7) has a very similar pattern to the observed total traffic counts in Lyon County.

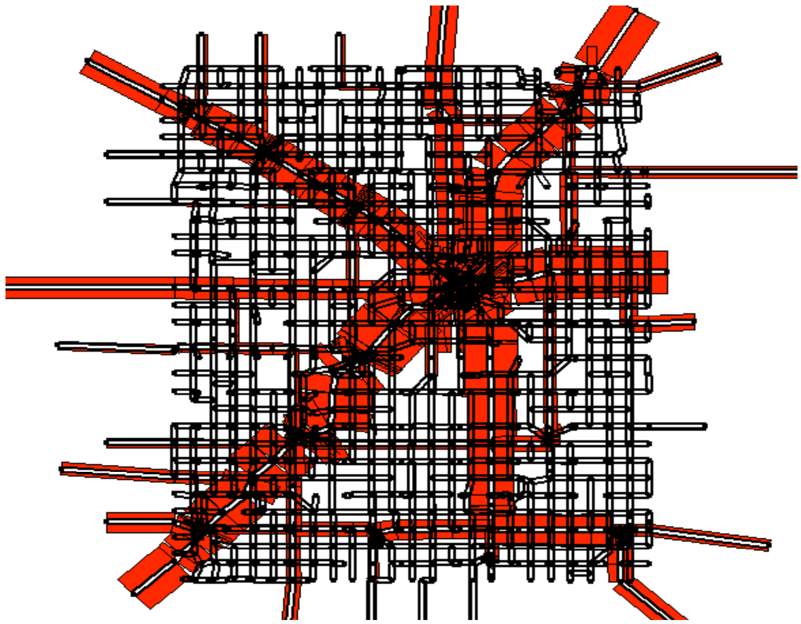

Figure 7: Truck volume map from Model (Scenario 1-without SLR)

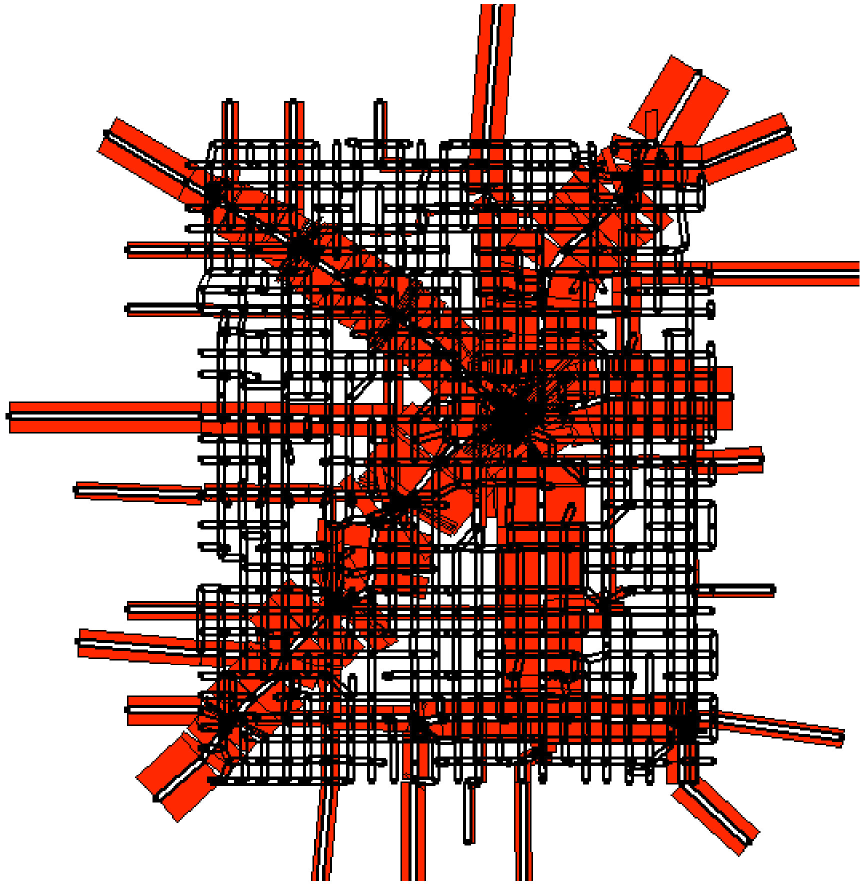

Figure 8: Truck volume map from Model (Scenario 2-with SLR)

Another method is to compare observed truck traffic on links outside the SLR period to their counterpart link counts in the model. Appendix 6 summarizes all the sites that reported data during the no SLR period and their modeled truck average daily traffic during the 10 months without SLR.

Figure 9 compares the model with observations. It can be seen from the plot that the two data sets have a strong linear relationship. In order to find how close the model reflects the actual conditions, several models were tried to fit the data. A linear model, quadratic model and Cobb-Douglas model were considered. But since the data contains zero values, the Cobb-Douglas model was abandoned. The linear model and quadratic model are chosen to fit the model. It is worthy of note that a zero intercept is assumed to avoid negative traffic counts in both models.

Linear Model:

$$Y = \beta_1 * x$$

Quadratic Model:

$$Y = \beta_1 * x + \beta_2 * x^2$$

Y: actual counts

x: model counts

β_1, β_2 : model coefficient

It can be seen from Table 11 that both the linear model and quadratic model have good R^2 values. The quadratic model has a little higher R^2 (0.845) than linear model

(0.837). But the *p* value of quadratic term in the quadratic model is higher than 0.05, which indicates an insignificant quadratic term. Thus, the linear model is adopted to calibrate the model results. Therefore, the traffic volume for each scenario was scaled up by a factor of 1.26. Table 12 gives the adjusted VKT with and without SLR for the three counties. A 26% adjustment factor was also applied to Clay and Olmsted County, for which there was inadequate data to conduct a localized calibration.

Table 11– Comparison of Linear and Quadratic model

Model	N	β_1	P value	β_2	P value	R^2
Linear	52	1.26	1.45E-26	N/A	N/A	0.837
Quadratic	52	1.535	2.66E-11	-0.00076	0.11338	0.845

Table 12 – Truck VKT from Base Runs

		Lyon County		Olmsted County		Clay County	
Scenario Number	Scenario	Raw VKT	Calibrated VKT	Raw VKT	Calibrated VKT	Raw VKT	Calibrated VKT
1	No SLR	83,184	104,812	264,420	333,169	182,708	230,212
2	With SLR	108,496	136,705	346,200	436,212	171,951	216,658
3 & 2’	Lift SLR on 7 and 9 only	102,594	129,268	320,100	403,326	181,983	229,299

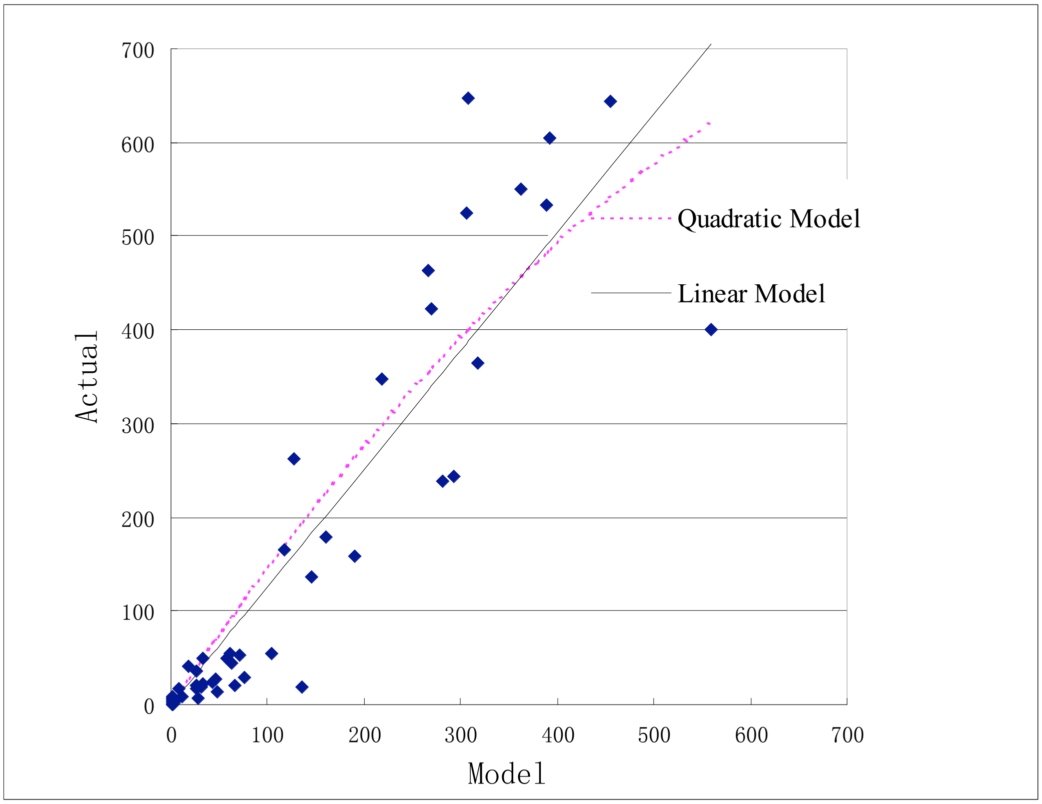

Figure 9: Plot of model vs. observed truck AADT for Lyon County

The freight demand model estimates truck volumes on each section of the roads under different policy scenarios. The model shows an increase of 30.4%, 30.9%, and 6.3% in truck distance traveled in Lyon, Olmsted, and Clay counties respectively if SLR is implemented strictly on all 5, 7, and 9-ton roads compared to the scenario without SLR. The model also concludes a 23.3%, 21.1%, and 5.8% increase of truck distance traveled in Lyon, Olmsted, and Clay counties respectively if restrictions are imposed only on 5-ton roads.

CHAPTER 5: CONCLUSIONS

In this thesis, a freight demand model for Minnesota Spring Load Restrictions policy is built to estimate the impacts of SLR on the freight transportation pattern. The model uses transportation planning techniques and allows various policy scenarios to be tested before being tested in practice.

The freight demand model calculates the truck trip generation rate of freight facilities in each county. Based on assumptions derived from the SLR survey, the trip distribution is set up. The truckers are assumed to choose the most economical mode for them to transport cargo. A shortest time path algorithm is applied in the route assignment step.

The freight demand model estimates truck volumes on each section of road under different policy scenarios. Observed truck counts and observed total traffic counts map are used to validate the model. A R-squared value of 0.836 shows a strong correlation between the model and observation.

The model shows an increase of 30.4%, 30.9%, and 6.3% in truck distance traveled in Lyon, Olmsted, and Clay counties respectively if SLR is implemented strictly on all 5, 7, and 9-ton roads compared to the scenario without SLR. The model also concludes a 23.3%, 21.1%, and 5.8% increase of truck distance traveled in Lyon, Olmsted, and Clay counties respectively if restrictions are imposed only on 5-ton roads. The results of the freight demand model is used in the further Benefit/Cost Analysis of Minnesota Spring Load Restrictions policy.

REFERENCES

Canadian Strategic Highway research Program. *Seasonal Load Restriction in Canada and Around the World.* Sept. 2000. http://www.cshrp.org/products/brief-21.pdf. Accessed Feb. 9, 2004.

Federal Highway Administration, US Department of Transportation, *Gravel Roads: Maintenance and Design Manual South Dakota Local Transportation Assistance Program*, November 2000, http://www.epa.gov/owow/nps/gravelroads/intro.pdf

Hashami, Maryam. *Operating costs for commercial vehicle operators in Minnesota.* Master's thesis of University of Minnesota. February 2004.

Isotalo, Jukka. Seasonal truck-load restrictions and road maintenance in countries with cold climate. The World Bank Report. March 1993. http://www.worldbank.org/transport/publicat/td-rd14.htm. Accessed Feb. 9, 2004.

ITE's Trip Generation Manual (7^{th} ed.). (2003). Washington, D. C: Institute of Transportation Engineers.

Iowa Department of Transportation, *1985 Truck Weight Index*

Li, Ning, Xi Zou, D. Levinson, *"Sharing Data Between Arc/Info and Emme/2: A practice in Lyon County, Minnesota",* proceedings at Transportation Research Forum's 2004 Annual Forum, Evanston, Illinois March 21-23

Levinson, D., Ning Li, Xi Zou, Brian Smalkoski, Maryam Hashami, *A Framework for*

Analyzing the Effects of Spring Load Restriction Journal of the Transportation Research Board (in press).

Minnesota Department of Transportation, *Restricted Gross Weight Table,* 2004.

Minnesota Department of Public Safety, *Seasonal weight restriction*, 2003 http://www.mrr.dot.state.mn.us/research/seasonal_load_limits/owpindex/owp.asp

Mirjam H.E, Wilhelm J, and Lori A. *Freight trip generation by firms. Paper for the 42nd European Congress of Regional Science Association*, Dortmund, 2002

National Agriculture Statistics Service, *grain production of Minnesota counties 2001*. http://www.nass.usda.gov/mn/agstat02/p097118.pdf

Public works of Lyon County. *Lyon load restriction map (2003) and traffic volume map (2000)*. http://www.lyonco.org/dept/pw/compplan.html. Accessed Feb. 15, 2004.

Public Works of Lyon County. *Lyon load restriction map (2003)* and *traffic volume map (2000)*. http://www.lyonco.org/dept/pw/compplan.html.

Refsdal, Geir. *The lifting of axle load restrictions during spring thaw- A Norwegian experiment.* Fifth International Conference on the Bearing Capacity of Roads airfields. Trondheim, Norway, 6-8 July 1998.

Refsdal, Senstad and Soerlie, *The lifting of all seasonal load restrictions in Norway in 1995-Background and Effects*, Transportation Research Board Annual Meeting, Washing DC, USA, 2004.

Smalkoski, Brian Richard. *Value of time for commercial vehicles operators in Minnesota*.

Master's thesis of University of Minnesota. December 2003.

Smalkoski, Brian, *Truck Trip Generation Model in Minnesota (working paper),* 2004.

Trimac Consulting Services Canada. *Evaluation of truck traffic patterns and impacts. Technical report for transport Canada Surface Policy and Program*. September 15,1999. http://www.tc.gc.ca/pol/en/Report/truckGrain1999/C4.htm. Accessed Feb. 9, 2004.

APPENDIX 1:

Matlab program of converting GIS map to Emme/2 format

The main purpose of this program is to convert a generate data file derived from GIS map to Emme/2 format. The input of this program is the generate data file "smp4lyon.m", which record the x,y coordinates of from node , to node and some vertex of each arc. The "lenghvol" file records the length and traffic counts on each link. The output of this program is the Lyon County map in Emme/2 format.

Program code:

```
clear;

format long;

% lengthvol consists of length and traffic count of each link

load lengthvol;

% the data file generated by ARC/Info

 filename='smp4lyon.m';


% Open Shape file

fd1=fopen(filename,'rb');

fd2=fopen('lyonnode','w');

fprintf(fd2,'t nodes init\n');

fd3=fopen('lyonlink','w');

fprintf(fd3,'t links init\n');

fd4=fopen('Linkwithmode','w');


% Variable define

fileEnd=0;

NodeNumber =0;

Node=[];

LinkNumber =0;

LinkSize=0;

LinkLength=0;
```

```
writelink=[];
link(1,1).x=0;
link(1,2).x=0;
numberEnd=0;

% assign 1.5 lane for link with traffic count 1000-2000, 2 lane for links with traffic count
more than 2000
[p,q]=size(lengthvol);
for i=1:p
   if lengthvol(i,2)>=1000
      if lengthvol(i,2)<2000
         lengthvol(i,3)=1.5;
      else
         lengthvol(i,3)=2;
      end
   else
        lengthvol(i,3)=1;
   end
end

% Main Loop
while fileEnd==0
   a=fgetl(fd1);
   A = sscanf(a,'%f');
   if ~isempty(A)
      %if it is not 'END'
      numberEnd= 0;
      if max(size(A))~=1
         % if it's coordinate, read x and y
         LinkSize = LinkSize +1;
         link(LinkNumber,LinkSize).x = A(1);
```

```
        link(LinkNumber,LinkSize).y = A(2);
      else
        % if it is not a coordinate
        % it is a beginning of a new link
        newLink=1;
        LinkLength =0;
        LinkSize =0;
        LinkNumber = LinkNumber + 1;
        %LinkNumber
      end
    else
      % it is the end of a link, record the from-node and to-node of this link
      xtmp=link(LinkNumber,1).x;
      ytmp=link(LinkNumber,1).y;
      if ~isempty(Node)
        fromNode = find(Node(:,1)==xtmp & Node(:,2) == ytmp);
        if isempty(fromNode)
          NodeNumber =NodeNumber +1;
          fromNode = NodeNumber;
          %NodeNumber
          Node(NodeNumber,1)=xtmp;
          Node(NodeNumber,2)=ytmp;
          fprintf(fd2,'a    %d    %6.0f    %6.0f    0    0    0    %4d\n',NodeNumber,
Node(NodeNumber,1), Node(NodeNumber,2), NodeNumber);
        end
      else
        fromNode=1;
        NodeNumber =NodeNumber +1;
        %NodeNumber
        Node(NodeNumber,1)=xtmp;
        Node(NodeNumber,2)=ytmp;
```

```
    fprintf(fd2,'a %d %6.0f %6.0f 0 0 0 %4d\n',NodeNumber, Node(NodeNumber,1),
Node(NodeNumber,2), NodeNumber);
  end
  xtmp=link(LinkNumber,LinkSize).x;
  ytmp=link(LinkNumber,LinkSize).y;
  toNode = find(Node(:,1)==xtmp & Node(:,2) == ytmp);
  if isempty(toNode)
    NodeNumber =NodeNumber +1;
    toNode= NodeNumber;
    %NodeNumber
    Node(NodeNumber,1)=xtmp;
    Node(NodeNumber,2)=ytmp;
    fprintf(fd2,'a %d %6.0f %6.0f 0 0 0 %4d\n',NodeNumber, Node(NodeNumber,1),
Node(NodeNumber,2), NodeNumber);
  end

  LinkLength= lengthvol(LinkNumber, 1)/5280;
  LinkVolume= lengthvol(LinkNumber, 2);
  LinklaneNumber=lengthvol(LinkNumber,3);
  LinkNumber;

  if ~isempty(writelink)
    addlink = find(writelink(:,1)==fromNode & writelink(:,2) == toNode);% |
writelink(:,1)==toNode & writelink(:,2) ==fromNode);
    if isempty(addlink)
      writelink(LinkNumber,1:2)=[fromNode toNode];

Linkdata(LinkNumber,1:6)=[LinkNumber,fromNode,toNode,LinkLength,LinklaneNumb
er,LinkVolume];
       if Linkdata(LinkNumber,6)>=1000
        Linkdata(LinkNumber,7)=3;
        mode='clmh';
       else
```

```
            if Linkdata(LinkNumber,6)>100
              Linkdata(LinkNumber,7)=1;
              mode='cl';
            else
              Linkdata(LinkNumber,7)=0;
              mode='c';
           end
         end
        fprintf(fd3,'a %d %d %4.3f   %s      1   %2.1f     5    %d    0    0\n', fromNode, toNode, LinkLength,mode,LinklaneNumber, LinkVolume);
        fprintf(fd4,'%d          %d                  %4.3f          %d           %2.1f           %d \n',fromNode,toNode,LinkLength,Linkdata(LinkNumber,7),LinklaneNumber,LinkVolume);


     else

Linkdata(LinkNumber,1:6)=[LinkNumber,fromNode,toNode,LinkLength,LinklaneNumber,LinkVolume];
        if Linkdata(LinkNumber,6)>=1000
         Linkdata(LinkNumber,7)=3;
         mode='clmh';
        else
            if Linkdata(LinkNumber,6)>100
              Linkdata(LinkNumber,7)=1;
              mode='cl';
            else
              Linkdata(LinkNumber,7)=0;
              mode='c';


             end


        end
```

```
fprintf(fd3,'a %d %d %4.3f %s 1 %2.1f 5 %d 0 0\n', toNode,fromNode,LinkLength,mode,LinklaneNumber, LinkVolume);
fprintf(fd4,'%d %d %4.3f %d %2.1f %d \n',toNode,fromNode,LinkLength,Linkdata(LinkNumber,7),LinklaneNumber,LinkVolume);



      end
    else
      writelink(LinkNumber,1:2)=[fromNode toNode];

Linkdata(LinkNumber,1:6)=[LinkNumber,fromNode,toNode,LinkLength,LinklaneNumber,LinkVolume];
      if Linkdata(LinkNumber,6)>=1000
         Linkdata(LinkNumber,7)=3;
          mode='clmh';
      else
          if Linkdata(LinkNumber,6)>100
             Linkdata(LinkNumber,7)=1;
             mode='cl';
          else
             mode='c';
             Linkdata(LinkNumber,7)=0
          end
      end

fprintf(fd3,'a %d %d %4.3f %s 1 %2.1f 5 %d 0 0\n', fromNode, toNode,LinkLength,mode,LinklaneNumber,LinkVolume);
fprintf(fd4,'%d %d %4.3f %d %2.1f %d \n',fromNode,toNode,LinkLength,Linkdata(LinkNumber,7),LinklaneNumber,LinkVolume);
```

```
        numberEnd = numberEnd + 1;
        if numberEnd ==2

            fileEnd=1;
            break;
        end
    end
end
fclose(fd1);
fclose(fd2);
fclose(fd3);
fclose(fd4);
```

\

APPENDIX 2:

Matlab Program of adding centroids to the Emme/2 map.

Program description:

The main purpose of this program is to add centroids to the Emme/2 map. Each freight facility is located to the nearest centroid. Some external points which connect the border of Lyon County are added. Connecter links which connect centroid and the nearest regular node are added. The input files are coordinates of each freight facility, network link data, link mode data. The output files are centroid number associated with each freight facility.

Program code:

```
clear;
load testdata;
load -ASCII chemdist.txt;
load -ASCII mafcoord.txt;
load -ASCII truckingfacility.txt;
load -ASCII retail.txt;
load -ASCII wholesale.txt;
load -ASCII outsideelevator.m
load Linkdata.mat
load Node.mat
load Linkwithmodenew

fd1=fopen('centroidlink','w');
fd2=fopen('lyonnode','a+');
fd3=fopen('lyonlink','a+');
fd4=fopen('outsideelevator','w')
fd5=fopen('insideelevator','w')
fd6=fopen('newchemnode','w')
fd7=fopen('newmalnode','w')
fd8=fopen('newtrucknode','w')
fd9=fopen('newretailnode','w')
```

```
fd10=fopen('newwholesale','w')
fd11=fopen('centroiddata','w')

% wholesale facility in lyon
bx=3.39149838;
ax=-427181.7413;
by=3.162369805;
ay=-15383939.54;
[m,n]=size(wholesale);
for i=1:m
   wholesalenew(i,1)=ax+bx*wholesale(i,1);
   wholesalenew(i,2)=ay+by*wholesale(i,2);
end
%retail facility in lyon
 %transform to new coordinates

bx=3.39149838;
ax=-427181.7413;
by=3.162369805;
ay=-15383939.54;
[m,n]=size(retail);
for i=1:m
    retailnew(i,1)=ax+bx*retail(i,1);
    retailnew(i,2)=ay+by*retail(i,2);
end
%trucking facility in lyon

%transform to new coordinates
bx=3.39149838;
ax=-427181.7413;
by=3.162369805;
```

```
ay=-15383939.54;
[m,n]=size(truckingfacility);
for i=1:m
    trucknew(i,1)=ax+bx*truckingfacility(i,1);
    trucknew(i,2)=ay+by*truckingfacility(i,2);
 end
%chemdistribution center in lyon
%transform to new coordinates
bx=3.39149838;
ax=-427181.7413;
by=3.162369805;
ay=-15383939.54;
[m,n]=size(chemdist);
for i=1:m
    chemnew(i,1)=ax+bx*chemdist(i,1);
    chemnew(i,2)=ay+by*chemdist(i,2);
 end
 %manufacturing facility in lyon
%transform to new coordinates
bx=3.39149838;
ax=-427181.7413;
by=3.162369805;
ay=-15383939.54;
[m,n]=size(mafcoord);
for i=1:m
    malnew(i,1)=ax+bx*mafcoord(i,1);
    malnew(i,2)=ay+by*mafcoord(i,2);

 end
```

```
% Elevators in Lyon county
lyon=[280939.13	4934079.01
264603.57	4911572.89
280907.08	4921551.55
290699.40	4901006.54
288012.03	4942967.51
270832.70	4901953.31
279446.40	4899738.63
270462.08	4932728.75
277901.66	4924932.16
284924.75	4910578.72
256715.60	4942246.33
287941.74	4942842.19];

%Elevators in counties nearby: Lincoln, Murray, Redwood, Yellow Medicine
othercounty=[249684.32	4907359.53
227782.15	4933734.24
246515.29	4919354.37
237611.12	4906029.75
241663.91	4928665.88
263308.63	4868198.19
286614.99	4882613.84
287811.90	4869636.56
276473.77	4866452.93
295753.80	4881117.44
263191.98	4875783.46
262635.25	4868044.51
271243.56	4876212.17
291075.02	4860683.39
279431.73	4874834.63
282783.97	4861780.25
```

```
320242.52	4918999.87
331721.54	4933269.30
318817.78	4900142.64
330534.56	4897222.09
323751.60	4909379.34
311177.78	4899458.09
336465.52	4915955.17
324341.39	4940584.67
320234.66	4919007.08
296729.81	4921727.21
311139.48	4899456.27
320192.27	4919067.46
302604.88	4899837.78
346309.93	4919952.53
307913.06	4930912.15
336317.45	4916139.10
329885.94	4897575.08
346278.37	4919981.62
277839.96	4963544.41
308283.53	4943172.29
240385.96	4955912.77
248790.63	4947237.76
298694.09	4947522.38
298054.14	4965166.87
292147.63	4952076.35
240447.01	4956414.95
278052.45	4963544.14];

[m n]=size(lyon);
bx=3.39149838;
ax=-427181.7413;
```

```
by=3.162369805;
ay=-15383939.54;

for i=1:m
    lyonnew(i,1)= ax+ bx*lyon(i,1);
    lyonnew(i,2)= ay+ by*lyon(i,2);
end

[m n]=size(othercounty);
for i=1:m
    othercountynew(i,1)= ax+ bx*othercounty(i,1);
    othercountynew(i,2)= ay+ by*othercounty(i,2);
end

lyon=lyonnew;
othercounty=othercountynew;
% add 225 centroid in the network

d0x=500
xmin=443309-d0x; %254804
xmax=558734+d0x; %294068
ymin=100389-d0x; %4896921;
ymax=258575+d0x; %4946518;
dx=(xmax-xmin)/30;
dy=(ymax-ymin)/30;

centr=2000;
numberCentroid=0;
for i=1:15
    for j=1:15
```

```
        centroid=centr+(i-1)*15+j;
        numberCentroid=numberCentroid+1;
        cen((i-1)*15+j,1:3)=[centroid xmin+i*(2*dx)-dx ymin+j*(2*dy)-dy];
        fprintf(fd2,'a* %d %6.0f %6.0f 0 0 0 %4d\n',centroid, cen((i-1)*15+j,2), cen((i-
1)*15+j,3), centroid);
    end
end

% Elevators in Lyon county: find the nearest centroid and record its node number and
coordinates to file insideelevator
[numbElevat tmp]=size(lyon);
dist=[];
for i=1:numbElevat
    xtmp=lyon(i,1);
    ytmp=lyon(i,2);
    for j=1:numberCentroid
        dist(j)=sqrt((cen(j,2)-xtmp)^2 +(cen(j,3)- ytmp)^2)/5280;
    end
    closeCentroid = find(dist(:)==min(dist));
        fprintf(fd5,'%d                                   %6.0f                                   %6.0f
\n',cen(closeCentroid,1),cen(closeCentroid,2),cen(closeCentroid,3));

end
localcentroid=numberCentroid;

totalelevator=numbElevat;
% Elevators outside Lyon county: load the external elevator and write it to lyonnode

[m,n]=size(outsideelevator);
for i=1:m
    fprintf(fd2,'a*    %d    %6.0f    %6.0f    0    0    0    %4d\n',outsideelevator(i,1),
outsideelevator(i,2),outsideelevator(i,3), outsideelevator(i,1));
```

```
 end
 % add chemidistribution centroid
[m,n]=size(chemnew)
InternalCentroid=225;
dist=[];

for i=1:m
    xtmp=chemnew(i,1);
    ytmp=chemnew(i,2);
    for j=1:InternalCentroid
        dist(j)=sqrt((cen(j,2)-xtmp)^2 +(cen(j,3)- ytmp)^2)/5280;
    end
  closeCentroid = find(dist(:)==min(dist));
        fprintf(fd6,'%d                        %6.0f                        %6.0f
%6.2f\n',cen(closeCentroid,1),cen(closeCentroid,2),cen(closeCentroid,3),chemdist(i,3));
end


% find the nearest centroid as manufacture facility centroid

[m,n]=size(malnew)
InternalCentroid=225
dist=[];
for i=1:m
    xtmp=malnew(i,1);
    ytmp=malnew(i,2);
    for j=1:InternalCentroid
        dist(j)=sqrt((cen(j,2)-xtmp)^2 +(cen(j,3)- ytmp)^2)/5280;
    end
    closeCentroid = find(dist(:)==min(dist));
        fprintf(fd7,'%d              %6.0f              %6.0f              %6.2f
```

```
\n',cen(closeCentroid,1),cen(closeCentroid,2),cen(closeCentroid,3),mafcoord(i,4));
    end

% Add trucking faciliy centroid
[m,n]=size(trucknew)
InternalCentroid=225
dist=[];
for i=1:m
    xtmp=trucknew(i,1);
    ytmp=trucknew(i,2);
    for j=1:InternalCentroid
        dist(j)=sqrt((cen(j,2)-xtmp)^2 +(cen(j,3)- ytmp)^2)/5280;
    end
   closeCentroid = find(dist(:)==min(dist));
        fprintf(fd8,'%d                                        %6.0f                                        %6.0f
%6.2f\n',cen(closeCentroid,1),cen(closeCentroid,2),cen(closeCentroid,3),truckingfacility(
i,4));


end
%Add retail facility centroid


[m,n]=size(retailnew)
InternalCentroid=225;
dist=[];
for i=1:m
    xtmp=retailnew(i,1);
    ytmp=retailnew(i,2);
    for j=1:InternalCentroid
        dist(j)=sqrt((cen(j,2)-xtmp)^2 +(cen(j,3)- ytmp)^2)/5280;
    end
  closeCentroid = find(dist(:)==min(dist));
```

```
    fprintf(fd9,'%d                    %6.0f                    %6.0f
%6.2f\n',cen(closeCentroid,1),cen(closeCentroid,2),cen(closeCentroid,3),retail(i,4));

end
% Add wholesale facility centroid
[m,n]=size(wholesalenew)
InternalCentroid=225;
dist=[];
for i=1:m
  xtmp=wholesalenew(i,1);
  ytmp=wholesalenew(i,2);
  for j=1:InternalCentroid
    dist(j)=sqrt((cen(j,2)-xtmp)^2 +(cen(j,3)- ytmp)^2)/5280;
  end
  closeCentroid = find(dist(:)==min(dist));
    fprintf(fd10,'%d              %6.0f              %6.0f              %6.2f
\n',cen(closeCentroid,1),cen(closeCentroid,2),cen(closeCentroid,3),wholesale(i,4));

end
% Generate Links from internal centroids to nodes
type=0;

[m1,n1]=size(Linkwithmodenew);
for i=1:InternalCentroid
  Linknum=0;
  type=0;
  dist=[];
  Nodenum=[];
  x=cen(i,2);
  y=cen(i,3);
 Insidelink=[];
 dist1=[];
```

```
%select links that has node in the area of each centroid
for k=1:m1
    if x-0.5*dx<=Node(Linkwithmodenew(k,2),1)& Node(Linkwithmodenew(k,2),1)<=x+0.5*dx & y-0.5*dy<=Node(Linkwithmodenew(k,2),2)& Node(Linkwithmodenew(k,2),2)<=y+0.5*dy
        Linknum=Linknum+1;
        Insidelink(Linknum)=Linkwithmodenew(k,1);
    end
end

 [n2,m2]=size(Insidelink);
if m2>0
    count=0;
     for j=1:m2
      if Linkwithmodenew(Insidelink(j),5)>type
          type=Linkwithmodenew(Insidelink(j),5);

      end
     end



    type;
    Numoftype=0;

     for l=1:m2
      if Linkwithmodenew(Insidelink(l),5)==type
% calculate distance from centroid to these nodes and add link from centroid to the nearest one.
          Numoftype=Numoftype+1;
          Nodenum(Numoftype)=Linkwithmodenew(Insidelink(l),2);
          dist(Numoftype)=sqrt((cen(i,2)-Node(Linkwithmodenew(Insidelink(l),2),1))^2 +(cen(i,3)- Node(Linkwithmodenew(Insidelink(l),2),2))^2)/5280;
```

```
    end
  end
    closeNode=find(dist(:)==min(dist));
    Nodenum(closeNode(1));
    if type==3
      mode='clmh';
    else
      if type==2;
      mode='clm';
      else
        if type==1
          mode='cl';
        else
          mode='c';
         end
       end

     end

 fprintf(fd1,'a  %d  %d  %4.3f  %s       1    1    5     0     0      0\n',cen(i,1),
Nodenum(closeNode(1)),min(dist),mode);

 fprintf(fd1,'a  %d  %d  %4.3f  %s        1     1     5      0      0       0\n',
Nodenum(closeNode(1)),cen(i,1),min(dist),mode);
  %  fprintf(fd11,'%d  %d  %4.3f  %d       1       5     0     0      0\n',cen(i,1),
Nodenum(closeNode(1)),min(dist),type);
 else
  for r=1:NodeNumber
    dist1(r)=sqrt((Node(r,1)-cen(i,2))^2 +(Node(r,2)- cen(i,3))^2)/5280;
  end
  closeNodes= find(dist1(:)==min(dist1(:)));
  NearLink=find(Linkwithmodenew(:,2)==closeNodes(1));
```

```
        [m3,n3]=size(NearLink);
        type2=0;
        for z=1:m3
          if Linkwithmodenew(NearLink(z),5)>type2
              type2=Linkwithmodenew(NearLink(z),5);
          end
        end
       if type2==3
              mode='clmh';
          else
              if type2==2;
              mode='clm';
              else
                 if type2==1
                   mode='cl';
                 else
                   mode='c';
                  end
               end


          end



        fprintf(fd1,'a  %d  %d  %4.3f  %s        1    1    5     0     0       0\n',cen(i,1),
closeNodes(1),min(dist1),mode);


        fprintf(fd1,'a   %d   %d   %4.3f   %s          1      1      5        0         0
0\n',closeNodes(1),cen(i,1),min(dist1),mode);
        %fprintf(fd11,'%d  %d  %4.3f  %d        1       5      0      0       0\n',cen(i,1),
closeNodes(1),min(dist1),type2);
```

```
    end

  end

% Generate Links from external centroids to real expors on the boundary

[m,n]=size(outsideelevator);
 for i=1:m
  dist(i)=sqrt((Node(outsideelevator(i,4),1)-outsideelevator(i,2))^2
+(Node(outsideelevator(i,4),2)- outsideelevator(i,3))^2)/5280;
  %fprintf(fd3,'a %d %d %4.3f   clmh      1   1     5    0    0    0\n', outsideelevator(i,1),
outsideelevator(i,4), dist(i));
  %fprintf(fd3,'a %d %d %4.3f   clmh      1   1     5    0    0    0\n', outsideelevator(i,4),
outsideelevator(i,1), dist(i));
  %fprintf(fd11,'%d  %d  %4.3f   3      1   1      5     0     0     0\n', outsideelevator(i,1),
outsideelevator(i,4), dist(i));

  end
fclose(fd1);
fclose(fd2);
fclose(fd3);
fclose(fd4);
fclose(fd5);
fclose(fd6);
fclose(fd7);
fclose(fd8);
fclose(fd9);
fclose(fd10);
fclose(fd11);
```

APPENDIX 3:

Matlab program of generating demand matrix in NoSLR scenario

Program description:

The main objective of this program is to generate Origin-Destination (OD) matrix between centroids in NoSLR scenario.

Program code:

```
clear;
change5=1;
change7=1;
tic
format long;
load -ASCII insideelevator.m;
load -ASCII outsideelevator.m;
load -ASCII newchemnode;
load -ASCII newtrucknode;
load -ASCII exteeleformal.m;
load -ASCII newmalnode;
load -ASCII newretailnode;
load -ASCII newwholesale;
load -ASCII newotherfreight;
load -ASCII centroiddata.m;
load cen.mat
load cdata.mat

% The size of matrix is derived from the d211. we have 250 centroids in the road network
M0=zeros(250,250);
M1=zeros(250,250);
M2=zeros(250,250);
M3=zeros(250,250);
```

```
% Centroids are labeled with id number starting with 2000.
Centr=2000;

% all centroids
% Total Trip from farms
%  trip from each farm to internal elevator
% Tirp distribution between farm and internal elevator
% Each farm will find the nearest grain elevators.

[m1 n1]=size(insideelevator);
origin=cen(1:225,1:3);
dest1=insideelevator(1:m1, 1:3);
[Osize tmp]=size(origin);
[Dsize tmp]= size(dest1);
count1=zeros(Dsize,2);
dij=[];
for i=1:Osize
   for j=1:Dsize
      dij(j) =sqrt((origin(i,2)-dest1(j,2))^2 +(origin(i,3)-dest1(j,3))^2 )/5280;
   end
   mindist=min(dij);
   closeelevator=find(dij(:)==mindist);
   close=closeelevator(1,1);
       min(centroiddata(i,4),centroiddata(dest1(close,1)-Centr,4));
      if  min(centroiddata(i,4),centroiddata(dest1(close,1)-Centr,4))==0
           Pi=0.67;
      M0(i,dest1(close,1)-2000)=M0(i,dest1(close,1)-2000)+Pi;
   else
      if  min(centroiddata(i,4),centroiddata(dest1(close,1)-Centr,4))==1
            Pi=0.67;
       M1(i,dest1(close,1)-                                2000)=M1(i,dest1(close,1)-2000)+Pi;
```

```
        else
          if min(centroiddata(i,4),centroiddata(dest1(close,1)-Centr,4))==2
              Pi=0.67;
          M2(i,dest1(close,1)-2000)=M2(i,dest1(close,1)-2000)+Pi;
          else
              Pi=0.67;
               M3(i,dest1(close,1)-2000)=M3(i,dest1(close,1)-2000)+Pi;
          end
        end
    end
   % return trip are all empty truck which can go on any road
   M0(dest1(close,1)-2000,i)=M0(dest1(close,1)-2000,i)+Pi;
   % calculating how many trip each grain elevator attracts
   count1(close,1)=count1(close,1)+1;
   count1(close,2)=count1(close,2)+Pi;

end
sum1=0;
for i=1:250
    for j=1:250
     sum1=sum1+M0(i,j)+M1(i,j)+M2(i,j)+M3(i,j);
 end
end
sum1

% find the nearest chem distribution center for each farm, truck goes from nearest chem
distribution center to the farm
[m1 n1]=size(newchemnode);
origin=cen(1:225,1:3);
dest1=newchemnode(1:m1, 1:3);
[Osize tmp]=size(origin);
```

```
[Dsize tmp]= size(dest1);
count=zeros(Dsize,1);
dij=[];
for i=1:Osize
   for j=1:Dsize
       dij(j) =sqrt((origin(i,2)-dest1(j,2))^2 +(origin(i,3)-dest1(j,3))^2 )/5280;
   end
   mindist=min(dij);
   closechem=find(dij(:)==mindist);
   close=closechem(1,1);
   count(close)=count(close)+1;
   od(i)=close;
   end
%trip distribution
for i=1:Osize
    demand(i)=newchemnode(od(i),4)/count(od(i));

      if min(centroiddata(i,4),centroiddata(dest1(od(i),1)-2000,4))==0
       M0(dest1(od(i),1)-2000,i)=M0(dest1(od(i),1)-2000,i)+demand(i);
   else
       if min(centroiddata(i,4),centroiddata(dest1(od(i),1)-2000,4))==1
         M1(dest1(od(i),1)-2000,i)=M1(dest1(od(i),1)-2000,i)+demand(i);

       else
           if min(centroiddata(i,4),centroiddata(dest1(od(i),1)-2000,4))==2
           M2(dest1(od(i),1)-2000,i)=M2(dest1(od(i),1)-2000,i)+demand(i);
           else
           M3(dest1(od(i),1)-2000,i)=M3(dest1(od(i),1)-2000,i)+demand(i);
           end
       end
   end
```

```
% for return trip, they are all empty
  M0(i,dest1(od(i),1)-2000)=M0(i,dest1(od(i),1)-2000)+demand(i);
    end



sum2=0;
for i=1:250
   for j=1:250
    sum2=sum2+M0(i,j)+M1(i,j)+M2(i,j)+M3(i,j);
 end
end
sum2

%trip from Manufacturing to few external points,doulbe direction,the demand is assigned
to each external points in proportion to real traffic counts.
origin=newmalnode;
dest1=outsideelevator;
[Osize tmp]=size(origin);
[Dsize tmp]= size(dest1);
dij=[];
 for i=1:Osize
    for j=1:Dsize

  if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==0
   M0(origin(i,1)-Centr,dest1(j,1)-Centr)=M0(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
   M0(dest1(j,1)-Centr,origin(i,1)-Centr)=M0(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
  else
      if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==1
        M1(origin(i,1)-Centr,dest1(j,1)-Centr)=M1(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
```

```
        M1(dest1(j,1)-Centr,origin(i,1)-Centr)=M1(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
      else
         if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==2
        M2(origin(i,1)-Centr,dest1(j,1)-Centr)=M2(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
        M2(dest1(j,1)-Centr,origin(i,1)-Centr)=M2(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
         else
         M3(origin(i,1)-Centr,dest1(j,1)-Centr)=M3(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
        M3(dest1(j,1)-Centr,origin(i,1)-Centr)=M3(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
         end
      end
   end


end
end
  sum3=0;


for i=1:250
    for j=1:250
     sum3=sum3+M0(i,j)+M1(i,j)+M2(i,j)+M3(i,j);
 end
end


sum3
 % trip from wholesale to external few points double direction
origin=newwholesale;
dest1=outsideelevator;
[Osize tmp]=size(origin);
```

```
[Dsize tmp]= size(dest1);
dij=[];
 for i=1:Osize
    for j=1:Dsize


  if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==0
    M0(origin(i,1)-Centr,dest1(j,1)-Centr)=M0(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
    M0(dest1(j,1)-Centr,origin(i,1)-Centr)=M0(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
   else
       if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==1
         M1(origin(i,1)-Centr,dest1(j,1)-Centr)=M1(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
         M1(dest1(j,1)-Centr,origin(i,1)-Centr)=M1(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
       else
           if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==2
         M2(origin(i,1)-Centr,dest1(j,1)-Centr)=M2(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
         M2(dest1(j,1)-Centr,origin(i,1)-Centr)=M2(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
           else
          M3(origin(i,1)-Centr,dest1(j,1)-Centr)=M3(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
         M3(dest1(j,1)-Centr,origin(i,1)-Centr)=M3(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
           end
       end
   end
    end
end
   sum4=0;
```

```
for i=1:250
    for j=1:250
     sum4=sum4+M0(i,j)+M1(i,j)+M2(i,j)+M3(i,j);
 end
end
sum4


% trip from retail to external points. (adopting previous method)
origin=newretailnode;
dest1=outsideelevator;
[Osize tmp]=size(origin);
[Dsize tmp]= size(dest1);
dij=[];
total=0;
for i=1:Dsize
 total=total+dest1(i,4);
end
  for i=1:Osize
    for j=1:Dsize


    if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==0
    M0(origin(i,1)-Centr,dest1(j,1)-Centr)=M0(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
    M0(dest1(j,1)-Centr,origin(i,1)-Centr)=M0(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
    else
       if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==1
    M1(origin(i,1)-Centr,dest1(j,1)-Centr)=M1(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
    M1(dest1(j,1)-Centr,origin(i,1)-Centr)=M1(dest1(j,1)-Centr,origin(i,1)-
```

```
Centr)+origin(i,4)*dest1(j,6);
       else
           if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==2
    M2(origin(i,1)-Centr,dest1(j,1)-Centr)=M2(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
    M2(dest1(j,1)-Centr,origin(i,1)-Centr)=M2(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
           else
    M3(origin(i,1)-Centr,dest1(j,1)-Centr)=M3(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
    M3(dest1(j,1)-Centr,origin(i,1)-Centr)=M3(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
           end
       end
   end
  end

end



sum5=0;


for i=1:250
    for j=1:250
     sum5=sum5+M0(i,j)+M1(i,j)+M2(i,j)+M3(i,j);
 end
end


sum5

  % trip from inside other freight facility  to external double direction (bidirectional)
```

```
origin=newotherfreight;
dest1=outsideelevator;
[Osize tmp]=size(origin);
[Dsize tmp]= size(dest1);
dij=[];
 for i=1:Osize
   for j=1:Dsize
     if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==0
   M0(origin(i,1)-Centr,dest1(j,1)-Centr)=M0(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
   M0(dest1(j,1)-Centr,origin(i,1)-Centr)=M0(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
  else
      if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==1
   M1(origin(i,1)-Centr,dest1(j,1)-Centr)=M1(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
   M1(dest1(j,1)-Centr,origin(i,1)-Centr)=M1(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
      else
          if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==2
   M2(origin(i,1)-Centr,dest1(j,1)-Centr)=M2(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
   M2(dest1(j,1)-Centr,origin(i,1)-Centr)=M2(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
          else
   M3(origin(i,1)-Centr,dest1(j,1)-Centr)=M3(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
   M3(dest1(j,1)-Centr,origin(i,1)-Centr)=M3(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);


          end
      end
  end
```

```
     end

end

sum55=0;

for i=1:250
    for j=1:250
     sum55=sum55+M0(i,j)+M1(i,j)+M2(i,j)+M3(i,j);
 end
end

sum55
%trip from trucking facility to external points
 origin=newtrucknode;
dest1=outsideelevator;
[Osize tmp]=size(origin);
[Dsize tmp]= size(dest1);
dij=[];
 for i=1:Osize
    for j=1:Dsize
       if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==0
    M0(origin(i,1)-Centr,dest1(j,1)-Centr)=M0(origin(i,1)-Centr,dest1(j,1)-Centr)+origin(i,4)*dest1(j,6);
    M0(dest1(j,1)-Centr,origin(i,1)-Centr)=M0(dest1(j,1)-Centr,origin(i,1)-Centr)+origin(i,4)*dest1(j,6);
   else
       if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==1
    M1(origin(i,1)-Centr,dest1(j,1)-Centr)=M1(origin(i,1)-Centr,dest1(j,1)-Centr)+origin(i,4)*dest1(j,6);
    M1(dest1(j,1)-Centr,origin(i,1)-
```

```
Centr)+origin(i,4)*dest1(j,6);
       else
           if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==2
    M2(origin(i,1)-Centr,dest1(j,1)-Centr)=M2(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
    M2(dest1(j,1)-Centr,origin(i,1)-Centr)=M2(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
           else
    M3(origin(i,1)-Centr,dest1(j,1)-Centr)=M3(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
    M3(dest1(j,1)-Centr,origin(i,1)-Centr)=M3(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);


           end
       end
   end


     end
    end


sum6=0;
for i=1:250
    for j=1:250
     sum6=sum6+M0(i,j)+M1(i,j)+M2(i,j)+M3(i,j);
 end
end


sum6
%trip from Grain elevator to few external points
%  assuming 60% going out and 60% going into
origin=insideelevator;
dest1=outsideelevator;
```

```
[Osize tmp]=size(origin);
[Dsize tmp]= size(dest1);
dij=[];
 for i=1:Osize
   for j=1:Dsize

  if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==0
   M0(origin(i,1)-Centr,dest1(j,1)-Centr)=M0(origin(i,1)-Centr,dest1(j,1)-Centr)+count1(i,2)*0.60*dest1(j,6)*1;
   M0(dest1(j,1)-Centr,origin(i,1)-Centr)=M0(dest1(j,1)-Centr,origin(i,1)-Centr)+count1(i,2)*0.60*dest1(j,6)*1;
  else
      if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==1
        M1(origin(i,1)-Centr,dest1(j,1)-Centr)=M1(origin(i,1)-Centr,dest1(j,1)-Centr)+count1(i,2)*0.60*dest1(j,6);
        M1(dest1(j,1)-Centr,origin(i,1)-Centr)=M1(dest1(j,1)-Centr,origin(i,1)-Centr)+count1(i,2)*0.60*dest1(j,6);
      else
         if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==2
        M2(origin(i,1)-Centr,dest1(j,1)-Centr)=M2(origin(i,1)-Centr,dest1(j,1)-Centr)+count1(i,2)*0.60*dest1(j,6);
        M2(dest1(j,1)-Centr,origin(i,1)-Centr)=M2(dest1(j,1)-Centr,origin(i,1)-Centr)+count1(i,2)*0.60*dest1(j,6);
         else
         M3(origin(i,1)-Centr,dest1(j,1)-Centr)=M3(origin(i,1)-Centr,dest1(j,1)-Centr)+count1(i,2)*0.60*dest1(j,6);
         M3(dest1(j,1)-Centr,origin(i,1)-Centr)=M3(dest1(j,1)-Centr,origin(i,1)-Centr)+count1(i,2)*0.60*dest1(j,6);
       end
      end
  end

end
```

```
end
  sum7=0;

for i=1:250
    for j=1:250
     sum7=sum7+M0(i,j)+M1(i,j)+M2(i,j)+M3(i,j);
 end
end

sum7
```

% external to external traffic

% The fifith column is the ADT on the road lead to the external points, thus each direction should be half of the ADT.

```
origin=outsideelevator;
dest1=outsideelevator;
[Osize tmp]=size(origin);
[Dsize tmp]= size(dest1);
dij=[];
 for i=1:Osize
    for j=1:Dsize
   if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==0
    M0(origin(i,1)-Centr,dest1(j,1)-Centr)=M0(origin(i,1)-Centr,dest1(j,1)-Centr)+origin(i,5)*0.13*0.50*0.20*origin(j,6);

   else
     if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==1
         M1(origin(i,1)-Centr,dest1(j,1)-Centr)=M1(origin(i,1)-Centr,dest1(j,1)-Centr)+origin(i,5)*0.13*0.50*0.20*origin(j,6);
      else
          if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==2
         M2(origin(i,1)-Centr,dest1(j,1)-Centr)=M2(origin(i,1)-Centr,dest1(j,1)-
```

```
Centr)+origin(i,5)*0.13*0.50*0.20*origin(j,6);

         else
         M3(origin(i,1)-Centr,dest1(j,1)-Centr)=M3(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,5)*0.13*0.50*0.20*origin(j,6);
         end
         end
     end

    end
   end
sum8=0;

for i=1:250
    for j=1:250
     sum8=sum8+M0(i,j)+M1(i,j)+M2(i,j)+M3(i,j);
 end
end
sum8
    % % trip from Chemical distribution Center to external points
origin=newchemnode;
dest1=outsideelevator;
[Osize tmp]=size(origin);
[Dsize tmp]= size(dest1);
dij=[];
 for i=1:Osize
    for j=1:Dsize
        min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))
   if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==0
       M0(origin(i,1)-Centr,dest1(j,1)-Centr)=M0(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
```

```
      M0(dest1(j,1)-Centr,origin(i,1)-Centr)=M0(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
   else
      if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==1
        M1(origin(i,1)-Centr,dest1(j,1)-Centr)=M1(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
        M1(dest1(j,1)-Centr,origin(i,1)-Centr)=M1(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
      else
          if min(centroiddata(origin(i,1)-Centr,4),centroiddata(dest1(j,1)-Centr,4))==2
        M2(origin(i,1)-Centr,dest1(j,1)-Centr)=M2(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
        M2(dest1(j,1)-Centr,origin(i,1)-Centr)=M2(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
          else
        M3(origin(i,1)-Centr,dest1(j,1)-Centr)=M3(origin(i,1)-Centr,dest1(j,1)-
Centr)+origin(i,4)*dest1(j,6);
        M3(dest1(j,1)-Centr,origin(i,1)-Centr)=M3(dest1(j,1)-Centr,origin(i,1)-
Centr)+origin(i,4)*dest1(j,6);
          end
      end
   end


end
end
 sum9=0;
for i=1:250
   for j=1:250
    sum9=sum9+M0(i,j)+M1(i,j)+M2(i,j)+M3(i,j);
 end
end
sum9
    t=toc;
```

APPENDIX 4:

Matlab Program building demand matrix in format of Emme/2

Program description:

This program convert OD matrix into the format of Emme/2, which will be used as d311 file of Emme/2 databank.

Program code:

```
clear;
tic
load M0.mat;
load M1.mat;
load M2.mat;
load M3.mat;

fd1=fopen('MatrixJune23S5','a');
fprintf(fd1,'t matrices init   /@(#) d311.in 1.3@(#)\n');
fprintf(fd1,'a matrix=mf01 wod76d 0 olmsted 5 ton road demand\n');
centr=2000;
[Osize,Dsize]=size(M0);
for i=1:Osize
   fprintf(fd1,'%d ',i+centr);

   for j=1:Dsize
      fprintf(fd1,'%d:%1.5f ',j+centr, M0(i,j));
      if ~mod(j,5)
         fprintf(fd1,'\n');
         fprintf(fd1,'%d ',i+centr);
      end
   end
   fprintf(fd1,'\n');
```

```
end
fprintf(fd1,'a matrix=mf02 wod76d 0 olmsted 6 ton road demand\n');
centr=2000;
[Osize,Dsize]=size(M1);
for i=1:Osize
   fprintf(fd1,'%d ',i+centr);

   for j=1:Dsize
      fprintf(fd1,'%d:%1.5f ',j+centr, M1(i,j));
      if ~mod(j,5)
         fprintf(fd1,'\n');
         fprintf(fd1,'%d ',i+centr);
      end
   end
   fprintf(fd1,'\n');
end
fprintf(fd1,'a matrix=mf03 wod76d 0 olmsted 7 ton road demand\n');
centr=2000;
[Osize,Dsize]=size(M2);
for i=1:Osize
   fprintf(fd1,'%d ',i+centr);

   for j=1:Dsize
      fprintf(fd1,'%d:%1.5f ',j+centr, M2(i,j));
      if ~mod(j,5)
         fprintf(fd1,'\n');
         fprintf(fd1,'%d ',i+centr);
      end
   end
   fprintf(fd1,'\n');
end
```

```
fprintf(fd1,'a matrix=mf04 wod76d 0 olmsted 9,10 ton road demand\n');
centr=2000;
[Osize,Dsize]=size(M3);
for i=1:Osize
   fprintf(fd1,'%d ',i+centr);

   for j=1:Dsize
      fprintf(fd1,'%d:%1.5f ',j+centr, M3(i,j));
      if ~mod(j,5)
         fprintf(fd1,'\n');
         fprintf(fd1,'%d ',i+centr);
      end
   end
   fprintf(fd1,'\n');
end



fclose(fd1);
%fclose(fd2);
%fclose(fd3);
%fclose(fd4);
t=toc
```

APPENDIX 5:

Matlab program of calculating truck VKT in the road network

Program description:

The function of this program is to calculate the total truck VKT knowing truck volume on each section of the roads.

Program code:

```
a=[      ];
     fromnode=a(:,1);
     tonode=a(:,2);
     volume=a(:,3);
     time=a(:,4);
     length=a(:,5);
     evloume=a(:,3);

   [p,q]=size(a);
    for i=1:p
        if fromnode(i)<2000 & tonode(i)<2000
         evolume(i)=volume(i)+1.71;
     else
          evolume(i)=0;
      end

         if  evolume(i)==0;
             time(i)=0;
                  end
         end
    evolume1=evolume';
     VMTemme = sum(length.*evolume1)*1.6
```

APPENDIX 6:

Comparison of truck AADT to model under no SLR scenario.

Site	Road Category	AADT (trucks only)	Model
2536	5	0	1.71
2537	5	0	1.71
2540	5	8	1.71
2541	5	5	1.71
2569	5	3	1.71
2576	5	1	1.71
2578	5	6	1.71
2582	5	1	3.71
2584	5	4	3.71
2586	5	6	1.71
2589	5	3	3.71
2592	5	9	11.71
2530	7	24	43.71
2531	7	14	47.71
2532	7	17	25.71
2542	7	22	33.71
2543	7	49	57.71
2546	7	19	135.71
2562	7	35	25.71
2563	7	27	45.71
2566	7	20	25.71
2575	7	45	63.71
2577	7	20	25.71
2593	7	46	61.71
2539	9	41	17.71
2547	9	7	27.71
2549	9	21	65.71
2556	9	29	75.71
2565	9	54	103.71

2568	9	49	33.71
2580	9	17	7.71
2533	10	179	159.71
2535	10	533	389.71
2538	10	550	361.71
2544	10	244	293.71
2545	10	158	189.71
2548	10	605	391.71
2552	10	239	281.71
2557	10	365	317.71
2559	10	137	145.71
2560	10	165	117.71
2564	10	399.5	559.71
2570	10	19	31.71
2571	10	524	305.71
2581	10	643	455.71
2583	10	463	265.71
2587	10	52	71.71
2588	10	262	127.71
2591	10	347	217.71
2594	10	647	307.71
2595	10	55	61.71
2597	10	423	269.71