

**A LINEAR PROGRAMMING METHOD
FOR SYNTHESIZING ORIGIN-DESTINATION (O-D)
TRIP TABLES FROM TRAFFIC COUNTS
FOR INCONSISTENT SYSTEMS**

by

PENG LEI

A thesis submitted to the faculty of the
Virginia Polytechnic Institute and State University
In partial fulfillment of the requirements for the degree of
Master of Science
in
Industrial and Systems Engineering

APPROVED:

Dr. Hanif D. Sherali, Chairman

Dr. John Kobza

Dr. G. V. Loganathan

July, 1998
Blacksburg, Virginia

Key Word: Origin-Destination (OD) table, network flow, linear programming, equilibrium, object-oriented analysis

**A LINEAR PROGRAMMING METHOD
FOR SYNTHESIZING ORIGIN-DESTINATION (O-D)
TRIP TABLES FROM TRAFFIC COUNTS
FOR INCONSISTENT SYSTEMS**

by PENG LEI

Committee Chairman
Prof. Hanif D. Sherali
Department of Industrial and Systems Engineering

ABSTRACT

Origin-Destination (O-D) trip tables represent the demand-supply information of each directed zonal-pair in a given region during a given period of time. The effort of this research is to develop a linear programming methodology for estimating O-D trip tables based on observed link volumes. In order to emphasize the nature of uncertainty in the data and in the problem, the developed model permits the user's knowledge of path travel time to vary within a band-width of values, and accordingly modifies the user-optimality principle. The data on the observed flows might also not be complete and need not be perfectly matched. In addition, a prior trip table could also be specified in order to guide the updating process via the model solution. To avoid excessive computational demands required by a total numeration of all possible paths between each O-D pair, a Column Generation Algorithm (CGA) is adopted to exploit the special structures of the model. Based on the known capacity of each link, a simple formula is suggested to calculate the cost for the links having unknown volumes. An indexed cost is

introduced to avoid the consideration of unnecessary passing-through-zone paths, and an algorithm for solving the corresponding minimum-cost-path problem is developed. General principles on the design of an object-oriented code are presented, and some useful programming techniques are suggested for this special problem. Some test results on the related models are presented and compared, and different sensitivity analyses are performed based on different scenarios. Finally, several research topics are recommended for future research.

ACKNOWLEDGMENTS

The author is truly indebted to Dr. Hanif D. Sherali for his step-by-step guidance, encouragement and patience, and for his invaluable professional and personal assistance. The author also wishes to thank Dr. John Kobza and Dr. G. V. Loganathan for their consideration and for serving as committee members.

Special thanks are due to Eugene D. Arnold, Jr. of Virginia Transportation Research Council for his supportive efforts to provide the author with some helpful references. Thanks are also due to Dr. R. Sivanandan of the Center for Transportation Research, Virginia Tech, with whom the author discussed many topics used in this research.

The author is grateful to his mother and his wife, Manli Liu and Shuxun Li, for their invaluable contribution to this effort.

TABLE OF CONTENTS

ABSTRACT

ACKNOWLEDGMENT

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1	INTRODUCTION	1
1.1	Traffic Congestion Problem	1
1.2	Need for O-D Trip Tables	2
1.3	Research Goal and Outline of the Thesis	3
2	LITERATURE REVIEW	6
2.1	Conventional Analysis and Quick Response Models	6
2.2	Traffic Count-Based Models	7
	<i>Gravity-Based Models</i>	8
	<i>Equilibrium Models</i>	9
	<i>Entropy Models</i>	9
	<i>Statistical Models</i>	9
	<i>Neural Network Models</i>	10
	<i>Fuzzy Weight Models</i>	10
2.3	Equilibrium-Based Models	11
2.4	Linear Programming Models	14

2.5	Summary of Literature Review	17
3	PROBLEM ANALYSIS AND FORMULATION	19
3.1	Concepts and Notation	19
3.2	Model Formulation	23
3.3	Existence of Feasible Solutions	28
3.4	Using a Target Trip Table to Guide Solutions	33
3.5	Missing Volumes and Sub-System Optima	36
3.6	Summary and Comments	42
4	COLUMN GENERATION ALGORITHM AND INDEXED SHORTEST PATH PROBLEM	44
4.1	Limitations of the Standard Simplex Method	44
4.2	Decomposition Principle and Column Generation Technique	45
4.3	Choice of a Starting Basic Feasible Solution	46
4.4	Modified Column Generation Algorithm	49
	<i>Column Generation Procedure</i>	49
	<i>Summary of Column Generation Algorithm for Solving LPIM</i>	53
4.5	Column Generation Algorithm for Solving LPIM(TT)	55
4.6	Network Specification and a Shortest Path Algorithm	57
	<i>Network Restriction and Path Cost</i>	58
	<i>Indexed Path Cost</i>	58
	<i>Shortest Path Problem with Indexed Costs</i>	61
	<i>Linearization and Rescaling Indexed Costs</i>	65
5	OBJECT-ORIENTED ANALYSIS	68
5.1	Principles of the Design	68

5.2	Choice of Design Style and Programming Language	69
5.3	Object and Abstract Data Type	72
5.4	Template and Inheritance	76
5.5	Dynamic Link and Binding, Polymorphism	80
5.6	Summary and Technical Remarks	83
6	MODEL EVALUATION AND VALIDATION	87
6.1	Measures of Closeness Between O-D Trip Tables	87
6.2	Tests on a Sample Network	89
6.3	Tests on Real Networks	91
	<i>Networks and Models</i>	<i>91</i>
	<i>Pulaski Network</i>	<i>92</i>
	<i>Different Test Cases</i>	<i>93</i>
	<i>Sensitivity Analysis</i>	<i>96</i>
	<i>Comparison of Results</i>	<i>104</i>
6.4	Enhancing the Performance of Models	104
7	SUMMARY AND CONCLUSIONS	105
7.1	Summary	105
7.2	Recommendations for Future Research	106
	REFERENCES	108
	VITA	115

LIST OF FIGURES

Figure #		Page #
3.1	Flows are conserved at each intermediate node, but not consistent	22
3.2	Overflow, balanced and underflow node	28
3.3	Network that requires $\Lambda \geq U_{total}$	30
4.1	An ordinary network with costs measured in real numbers	59
4.2	The network of Figure 4.1, using the indexed link cost notation	60
4.3	A set of shortest paths using indexed costs	63
4.4	A solution of the shortest path problem to the network of Figure 4.1	64
6.1	Gur Network and its Characteristics	90
6.2	Street map and zones of Pulaski Town	94
6.3	Pulaski Network	95
TRIP TABLE AND VOLUME COMPARISONS FOR PULASKI NETWORK		
6.4	Case: Peak Hour, Measure: %RMSE(TT)	99
6.5	Case: Peak Hour, Measure: %MAE(TT)	99
6.6	Case: Peak Hour, Measure: PHI(TT)	100
6.7	Case: Peak Hour, Measure: %RMSE(VOL)	100
6.8	Case: Peak Hour, Measure: %RMSE(VOL)	101
6.9	Case: 24 Hour, Measure: %RMSE(TT)	101
6.10	Case: 24 Hour, Measure: %MAE(TT)	102
6.11	Case: 24 Hour, Measure: PHI(TT)	102
6.12	Case: 24 Hour, Measure: %RMSE(VOL)	103
6.13	Case: 24 Hour, Measure: %MAE(VOL)	103

LIST OF TABLES

Table #		Page #
6.1	Test results of the different models on the Gur network	91
6.2	Comparison using the Pulaski Network, Case: Peak Hour	97
6.3	Comparison using the Pulaski Network, Case: 24 Hour	97