

MODULE 3

3.1 Intervening Opportunities Model:

The gravity model is by far the most commonly used aggregate trip distribution model. But the gravity model does not exhaust all the theoretical possibilities. Intervening opportunities model which although much less used; offer real alternatives to the gravity model.

The basic idea behind the intervening-opportunities model is that trip making is not explicitly related to distance but to the relative accessibility of opportunities for satisfying the objective of the trip. The original proponent of this approach was Stouffer (1940), who also applied his ideas to migration and the location of services and residences. But it was Schneider (1959) who developed the theory in the way it is presented today.

Consider first a zone of origin i and rank all possible destinations in order of increasing distance from i . Then look at one origin-destination pair (i, j) , where j is the m^{th} destination in order of distance from i . There are $m-1$ alternative destinations actually closer (more accessible) from i . A trip maker would certainly consider those destinations as possible locations to satisfy the need giving rise to the journey: those are the intervening opportunities influencing a destination choice.

The basic hypothesis of this model given by Stouffer (1940) is that the number of trips from an origin zone to a destination zone is directly proportional to the number of opportunities at the destination zone and inversely proportional to the number of intervening opportunities. This hypothesis may be expressed as:

$$t_{ij} = (k \cdot a_j / v_j)$$

Where;

a_j = the total number of destination opportunities in zone j

v_j = the number of intervening destination opportunities between zones i and j

k = a proportionality constant to ensure that all trips with origins at zone i are distributed to destination opportunities.

Schneider (1959) has proposed the following modification of the Stouffer hypothesis. The Schneider hypothesis states that the probability that a trip will terminate in some volume of destination points is equal to the probability that this volume contains an acceptable destination times the probability that an acceptable destination closer to the origin of the trips has not been found. This hypothesis may be expressed as:

$$\text{pr}(dv) = [1 - \text{pr}(v)] \cdot l dv$$

Where;

$\text{pr}(dv)$ = the opportunity that a trip will terminate when dv destination opportunities are considered.

$\text{pr}(v)$ = the cumulative probability that a trip will terminate by time v possible destinations are considered = $(1 - k_i \exp(-lv))$

k_i = a constant for zone i to ensure that all the trips produced at zone i are distributed to zone i

v = the cumulative total of the destinations already considered.

l = a constant probability of a destination being accepted if it is considered.

The trip interchange between an i - j pair is given by:

$$t_{ij} = p_i (\text{probability of trip terminating in zone } j)$$

$$= p_i (\text{pr}(v_{j+1}) - \text{pr}(v_j))$$

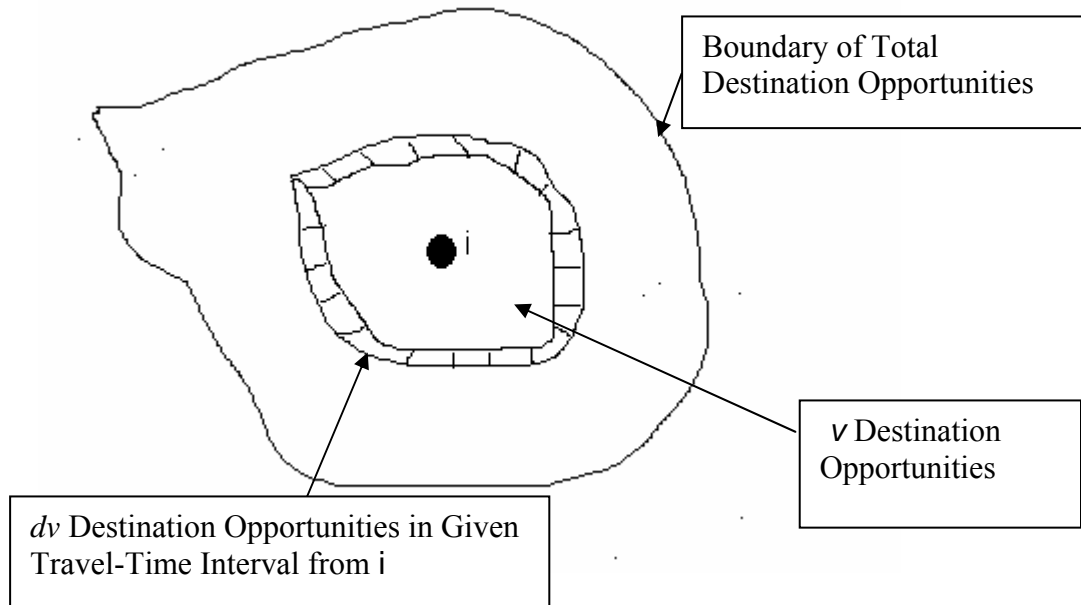
Where;

$\text{pr}(v_j)$ = the probability that a trip will have found a suitable destination in the opportunities already considered up to zone j

$\text{pr}(v_{j+1})$ = the probability that a trip will have found a suitable destination in the cumulative opportunities considered up to and including zone j .

From figure it may be restated as: the probability of locating within the dv opportunities being considered is equal to the product of the probability of not having located within the v opportunities already considered, and the probability of finding an

acceptable location within dv opportunities, given that a location has not already been found.



The intervening-opportunities model is interesting because it starts from different first principles in its derivation: it uses distance as an ordinal variable instead of a continuous cardinal one as in the gravity model. It explicitly considers the opportunities available to satisfy a trip purpose at increased distance from the origin. However, the model is not often used in practice, probably for the following reasons:

- The theoretical basis is less well known and possibly more difficult to understand by practitioners;
- The idea of matrices with destinations ranked by distance from the origin (the n^{th} cell for origin i is not destination n but the n^{th} destinations away from i) is more difficult to handle in practice;
- The theoretical and practical advantages of this function over the gravity model are not overwhelming;
- The lack of suitable software.

This is based on the concept that a trip cannot always go to the nearest destination and stop, it must consider the nearest destination and if that is not acceptable consider the next nearest and so on.

$$T_{ij} = \sum_{L_{\min}}^{L_{\max}} P_i (I - L)^U - (I - L)^{U_j - V_j}$$

T_{ij} = Total number of trips generated from zone I with a particular L value .

$(I-L)^U$ = Probability that a given trip will get to a given destination.

$(I - L)^{U_j - V_j}$ = Probability that a given trip will not find a satisfactory destination in j and go on to next zone.

L= Probability of a particular trip origin stopping at any randomly chosen destination.

V_j = Number of destinations found closer to the point of origin I than zone j.

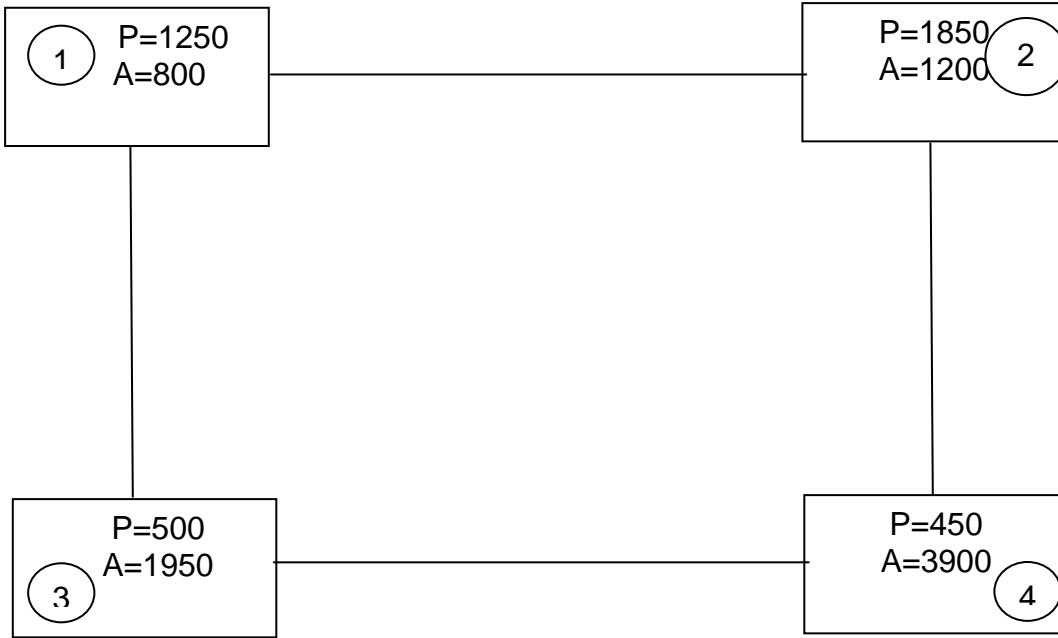
V_j = Number of destinations in zone j.

3.1.1 Example problem:

Find the trip interchange for the given figure?

P= Production

A=attraction



Iteration 1:

Zone	Productions	Attractions
1	1250	800
2	1850	1200
3	450	3900
4	500	1950

Let $l=0.00055$

$$T_{11} = 1250 (e^0 - e^{(-0.00055*800)}) = 445$$

$$T_{12} = 1250 (e^{(-0.00055*800)} - e^{(-0.00055*2000)}) = 389$$

$$T_{13} = 1250 (e^{(-0.00055*2000)} - e^{(-0.00055*5900)}) = 367$$

$$T_{14} = 1250 (e^{(-0.00055*5900)} - e^{(-0.00055*7850)}) = 32$$

Therefore

$$T_{1j} = 445 + 389 + 367 + 32 = 1233$$

$$T_{21} = 1850 (e^{(-1200*0.00055)} - e^{(-0.00055*2000)}) = 894$$

$$T_{22} = 1850 (e^0 - e^{(-0.00055*1200)}) = 340$$

$$T_{23} = 1850 (e^{(-0.00055*2000)} - e^{(-0.00055*5900)}) = 544$$

$$T_{24} = 1850(e^{-0.00055*5900} - e^{-0.00055*7850}) = 47$$

Therefore

$$\mathbf{T_{2j}=340+894+544+47=1825}$$

$$T_{31} = 450 (e^{-3900*0.00055} - e^{-0.00055*5100}) = 10$$

$$T_{32} = 450 (e^{-0.00055*5100} - e^{-0.00055*5900}) = 25$$

$$T_{33} = 450 (e^0 - e^{-0.00055*3900}) = 380$$

$$T_{34} = 450 (e^{-0.00055*5900} - e^{-0.00055*7850}) = 12$$

Therefore

$$\mathbf{T_{3j}=380+25+10+12=427}$$

$$T_{41} = 500 (e^{-1925*0.00055} - e^{-0.00055*3150}) = 84$$

$$T_{42} = 500 (e^{-0.00055*3150} - e^{-0.00055*3950}) = 31$$

$$T_{43} = 500 (e^{-0.00055*3950} - e^{-0.00055*7850}) = 50$$

$$T_{44} = 500 (e^0 - e^{-0.00055*1950}) = 329$$

Therefore

$$\mathbf{T_{4j}=84+31+50+329=494}$$

Values of l_1, l_2, l_3, l_4 :

$$l_i = \frac{1}{v_{ij} \log e} \cdot \log \frac{p_i^m}{p_i^m - t_{ij}} \quad (I=j)$$

Substituting the respective values

$$l_1=0.00059$$

$$l_2=0.00056$$

$$l_3=0.000565$$

$$l_4=0.00056$$

Using the modified values of l_{ij} , the next iteration is done.

Iteration 2:

$$L_1=0.00059$$

$$T_{11} = 1250 (e^0 - e^{(-0.00059*800)}) = 470$$

$$T_{12} = 1250 (e^{(-0.00059*800)} - e^{(-0.00059*2000)}) = 395$$

$$T_{13} = 1250 (e^{(-0.00059*2000)} - e^{(-0.00059*5900)}) = 345$$

$$T_{14} = 1250 (e^{(-0.00059*5900)} - e^{(-0.00059*7850)}) = 26$$

Therefore

$$\mathbf{T_{1j}=470+395+345+26=1236.}$$

$$T_{21} = 1850 (e^{(-1200*0.00056)} - e^{(-0.00056*2000)}) = 376$$

$$T_{22} = 1850 (e^0 - e^{(-0.00056*1200)}) = 905$$

$$T_{23} = 1850 (e^{(-0.00056*2000)} - e^{(-0.00056*5900)}) = 537$$

$$T_{24} = 1850 (e^{(-0.00056*5900)} - e^{(-0.00056*7850)}) = 45$$

Therefore

$$\mathbf{T_{2j}=905+376+537+45=1863}$$

$$T_{31} = 450 (e^{(-3900*0.000565)} - e^{(-0.000565*5100)}) = 24$$

$$T_{32} = 450 (e^{(-0.000565*5100)} - e^{(-0.000565*5900)}) = 9$$

$$T_{33} = 450 (e^0 - e^{(-0.000565*3900)}) = 400$$

$$T_{34} = 450 (e^{(-0.000565*5900)} - e^{(-0.000565*7850)}) = 10$$

Therefore

$$\mathbf{T_{3j}=24+9+400+10=443}$$

$$T_{41} = 500 (e^{(-1925*0.00056)} - e^{(-0.00056*3150)}) = 84$$

$$T_{42} = 500 (e^{(-0.00056*3150)} - e^{(-0.00056*3950)}) = 30$$

$$T_{43} = 500 (e^{(-0.00056*3950)} - e^{(-0.00056*7850)}) = 48$$

$$T_{44} = 500 (e^0 - e^{(-0.00056*1950)}) = 332$$

Therefore

$$\mathbf{T_{4j}=84+30+48+332=496}$$

Values of l_1, l_2, l_3, l_4 :

$$l_i = \frac{1}{v_{ij} \log e} \cdot \log \frac{p_i^m}{p_i^m - t_{ij}} \quad (I=j)$$

Substituting the respective values

$$l_1 = 0.000597$$

$$l_2 = 0.000565$$

$$l_3 = 0.000591$$

$$l_4 = 0.000567$$

Using the modified values of l_{ij} , the next iteration is done.

Iteration 3:

$$L_1 = 0.000597$$

$$T_{11} = 1250 (e^0 - e^{(-0.000597*800)}) = 474$$

$$T_{12} = 1250 (e^{(-0.000597*800)} - e^{(-0.000597*2000)}) = 396$$

$$T_{13} = 1250 (e^{(-0.000597*2000)} - e^{(-0.000597*5900)}) = 341$$

$$T_{14} = 1250 (e^{(-0.00059*5900)} - e^{(-0.00059*7850)}) = 25$$

Therefore

$$\mathbf{T_{1j} = 474 + 396 + 341 + 25 = 1236}$$

$$L_2 = 0.000565$$

$$T_{21} = 1850 (e^{(-1200*0.000565)} - e^{(-0.000565*2000)}) = 341$$

$$T_{22} = 1850 (e^0 - e^{(-0.000565*1200)}) = 905$$

$$T_{23} = 1850 (e^{(-0.000565*2000)} - e^{(-0.000565*5900)}) = 531$$

$$T_{24} = 1850 (e^{(-0.000565*5900)} - e^{(-0.000565*7850)}) = 44$$

Therefore

$$\mathbf{T_{2j}=341+905+531+44=1821}$$

$$L_3=0.000591$$

$$T_{31} = 450 (e^{(-3900*0.000591)} - e^{(-0.000591*5100)}) = 23$$

$$T_{32} = 450 (e^{(-0.000591*5100)} - e^{(-0.000591*5900)}) = 8$$

$$T_{33} = 450 (e^0 - e^{(-0.000591*3900)}) = 405$$

$$T_{34} = 450 (e^{(-0.000591*5900)} - e^{(-0.000591*7850)}) = 9$$

Therefore

$$\mathbf{T_{3j}=23+8+405+9=445}$$

$$L_4=0.000567$$

$$T_{41} = 500 (e^{(-1925*0.000567)} - e^{(-0.000567*3150)}) = 84$$

$$T_{42} = 500 (e^{(-0.000567*3150)} - e^{(-0.000567*3950)}) = 31$$

$$T_{43} = 500 (e^{(-0.000567*3950)} - e^{(-0.000567*7850)}) = 48$$

$$T_{44} = 500 (e^0 - e^{(-0.000567*1950)}) = 334$$

Therefore

$$\mathbf{T_{4j}=84+31+48+334=497}$$

Values of l_1, l_2, l_3, l_4 :

$$l_i = \frac{1}{v_{ij} \log e} \cdot \log \frac{P_i^m}{P_i^m - t_{ij}} \quad (I=j)$$

Substituting the respective values

$$l_1=0.000604$$

$$l_2=0.00057$$

$$l_3=0.000597$$

$$l_4=0.000567$$

Using the modified values of l_{ij} again, further iterations are carried out until consecutive l_i values are same.

3.2 Competing opportunities Model:

Tomazinis proposed the competing opportunities model as follows:

$$t_{ij} = p_i b_{ij} = p_i (pr_{aj})(pr_{sj})$$

where

pr_{aj} = the probability of attraction to zone j

= The destination opportunities in zone j divided by the sum of the destination opportunities available in time bands up to and including m

$$= \frac{a_j}{\sum_{x=1}^n a_x}$$

pr_{sj} = Probability of trip end allocation satisfaction in zone j

= 1 - the sum of the destination opportunities available in time bands up to n including band m divided by the sum of total destinations in the study area j

$$= 1 - \frac{\sum_{x=1}^m a_x}{\sum_{x=1}^n a_x}$$

x=any time band.

m=Time band into which zone j falls

a_x=The destination opportunities available in time band x

n= The last time band as measured from an origin zone i

3.3 Linear Programming Approach:

the objective is to minimize the total amount of travel time of trip makers in moving between origin and destination pairs . Blunden , Colston Blunden have formulated the trip distribution

The objective function is to minimize the total vehicle travel time:

$$\min \sum_i \sum_j \sum_k l(i) a(i, j, k) V(j, k)$$

Where

$l(i)$ =Travel time of link i

$V(j, k)$ =Volume of link i

$a(i, j, k)$ =1 if link i lies on the path j to k

=0, if not

Step 1: -Determining the basic feasible solution by Least Square Method.

Step 2: -There will be $3+2-1=4$ basic variables. Checking for optimality,

1. Only simplex method is used to determine the entry variable. If the optimality condition is satisfied, stop if not go to step 2.
2. Determine leaving variable using the simplex feasibility condition.

By the method of multipliers, for each basic variable:

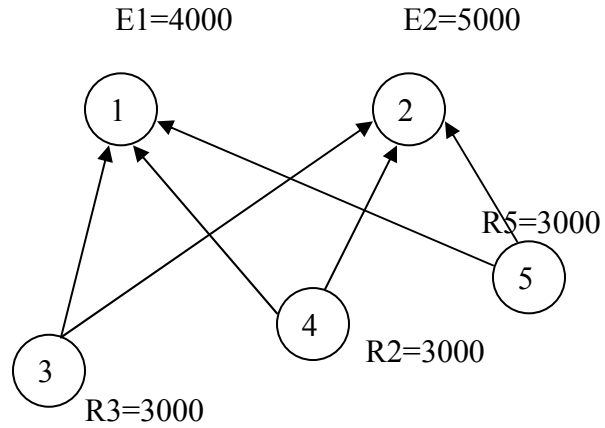
$u_i + v_j = c_{ij}$ Letting $u_3 = 0$, I can solve for the remaining as shown in table below

3.3.1 Example problem:

Assign the traffic to the various links for the network shown below.

Table. Travel Time Matrix

D \ O	1	2
3	11	22
4	16	18
5	21	10



The objective function is to minimize the total vehicle travel time:

$$\min \sum_i \sum_j \sum_k l(i).a(i, j, k).V(j, k)$$

Where

$l(i)$ =Travel time of link i

$V(j, k)$ =Volume of link i

$a(i, j, k)$ =1 if link i lies on the path j to k

=0, if not

subject to constraints

$$T_{31} + T_{32} = 3,000$$

$$T_{41} + T_{42} = 3,000$$

$$T_{51} + T_{52} = 3,000$$

$$T_{31} + T_{41} + T_{51} = 4,000$$

$$T_{32} + T_{42} + T_{52} = 5,000$$

As this is a balanced transportation problem, we can solve in using transportation label model. The travel times are in the N-E corner of the each table.

Step 1: -Determining the basic feasible solution by Least Square Method.

	1	2	
3	3000		3000
4	1000	2000	3000
5		3000	3000
	4000	5000	

1. Cell (3,1) has the least travel time. Assign 3000 to this cell. Row 1 is satisfied, satisfy column 1.
2. Next the cell with the least travel time is (5,2). Assign 3000 to this cell and adjust the total of column 2 to $5000-3000=2000$
3. Next cell with least travel time is (4,1) and allocate to it 1000
4. Allocate 2,000 to cell (4,2)

Step 2: -There will be $3+2-1=4$ basic variables. Checking for optimality,

3. Only simplex method is used to determine the entry variable. If the optimality condition is satisfied, stop if not go to step 2.
4. Determine leaving variable using the simplex feasibility condition.

By the method of multipliers, for each basic variable:

$u_i + v_j = c_{ij}$ Letting $u_3 = 0$, I can solve for the remaining as shown in table below

Basic Variable	(u, v)	Solution
(3,1)	$u_3 + v_1 = 11$	$v_1 = 11$
(4,1)	$u_4 + v_1 = 16$	$u_4 = 5$
(4,2)	$u_4 + v_2 = 18$	$v_2 = 13$
(5,2)	$u_5 + v_2 = 10$	$u_5 = -3$

I can use the tabulated solutions to get the values of the non-basic variables.

Non-basic Variable	$u_i + v_j - c_{ij}$	Comment
(3,2)	$u_3 + v_2 - c_{32} = 0 + 13 - 22 = -9$	The starting solution is optimal
(5,1)	$u_5 + v_1 - c_{51} = -3 + 11 - 21 = -13$	

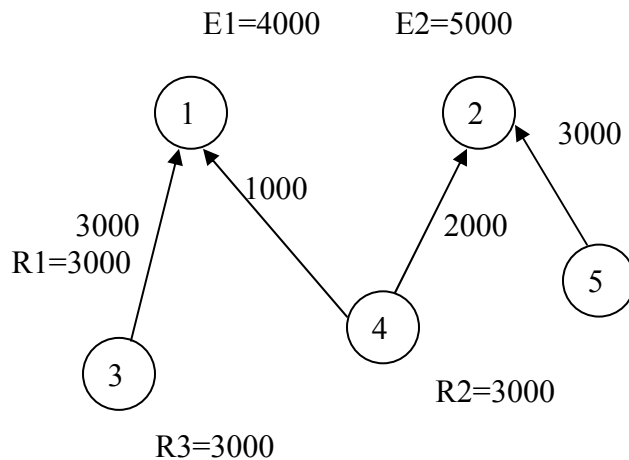
As the problem is minimization, the starting points are the solution as the non-basic variables has non-positive values.

The solutions are:

	T_{31}	T_{41}	T_{42}	T_{52}
Solution	3000	1000	2000	3000

The assigned trips can shown as:

D \ O	1	2
3	11	22
4	16	18
5	21	10



3.4 Wilson Modified Entropy Model:

The urban and regional scientist faces a number of theoretical problems. His activity is often a multi disciplinary one in the sense that he uses to concepts from several disciplines- economics, geography, and sociology and so on. The concept of entropy has, until recently, been used primarily in the non social science. It has hoped that 'entropy' enables the social scientist to take some of his basic problems in a fruitful way, and thus to make progress which might not be possible so easily which more orthodox tools.

Applications of Entropy:

1. It is used for theory building hypothesis (model) development.
2. Used in expressing the laws about system dynamics.
3. For interpretation procedures for the theories.

The main views of entropy is

1. The relationship of entropy to probability and uncertainty.
2. The entropy of probability distribution.
3. Entropy and Bayesian statistics.

Gravity Model

- Based on land use and transportation network
- Calibrated for many urban areas
- Simple
- Accurate
- Supported by the USDOT

$$T_{ij} = P_i \frac{A_j F_{ij} K_{ij}}{\sum A_j F_{ij} K_{ij}}$$

"The number of trips between 2 zones is directly proportional to the number of trip attractions at the destination zone and inversely proportional a function of the travel time"

T_{ij} = Trips produced in zone i and attracted to zone j

P_i = Trips produced in zone i

A_j = Trips attracted in zone j

F_{ij} = Friction factor for impedance (usually travel time) between zones i and j

K_{ij} = Socioeconomic adjustment factor for trips produced in i and attracted to j

How do we determine values for the variables?

- Recall Ps and As come from trip generation
- The sum of productions has to equal the sum of attractions

$$\sum P_i = \sum A_j$$

- Ks are used to force estimates to agree with observed trip interchanges (careful! do not use too many of these! Have a good reason for using them!)
- Fs are determined by a calibration process (by purpose), and depend upon the willingness of folks to make trips of certain lengths for certain purposes recall... trip purposes

HBW - home based work

HBO - home based other

NHB - non-home based

HBS - home based school

Derivation of Gravity model:

$$F_{ij} = \gamma \frac{m_1 m_2}{d_{12}^2}$$

$$T_{12} = k \frac{O_1 D_2}{d_{12}^2}$$

Double the values of γ , O_i , D_j

$$T_{12} = k \frac{O_1 D_2}{d_{12}^2}$$

Trip end balance is required

Balancing factors are A_i & B_j

Take the impedance in the form of generalized function

$$\mathcal{L} = \ln W(T_{ij}) + \lambda_i^{(1)} (O_i - \sum_j T_{ij}) + \lambda_j^{(2)} (D_j - \sum_i T_{ij}) + \beta (C - \sum_i \sum_j T_{ij} C_{ij})$$

$$T_{ij} = \exp(-\lambda_i^{(1)} - \lambda_j^{(2)} - \beta(C_{ij}))$$

$$\sum_j T_{ij} = O_i \text{ or } \exp(-\lambda_i^{(1)}) \sum_j \exp(-\lambda_j^{(2)} - \beta(C_{ij}))$$

$$\exp(-\lambda_i^{(1)}) = O_i / A_i \sum_j \exp(-\lambda_j^{(2)} - \beta(C_{ij}))^{-1}$$

$$\exp(-\lambda_j^{(2)}) = D_j / B_j \sum_i \exp(-\lambda_i^{(1)} - \beta(C_{ij}))^{-1}$$

$$T_{ij} = A_i O_j B_j D_i \exp(-\beta C_{ij})$$

$$A_i = \frac{\exp(-\gamma_i^1)}{O_i}$$

$$B_j = \frac{\exp(-\gamma_j^2)}{D_j}$$

$$A_i = \left[\sum_j D_j \exp(-\lambda_j^{(2)} - \beta(C_{ij}))^{-1} \right]$$

$$B_j = \left[\sum_i O_i \exp(-\lambda_i^{(1)} - \beta(C_{ij}))^{-1} \right]$$

$$A_i = \frac{1}{\sum_j B_j D_j \exp(-\beta C_{ij})}$$

3.5 Trip Assignment:

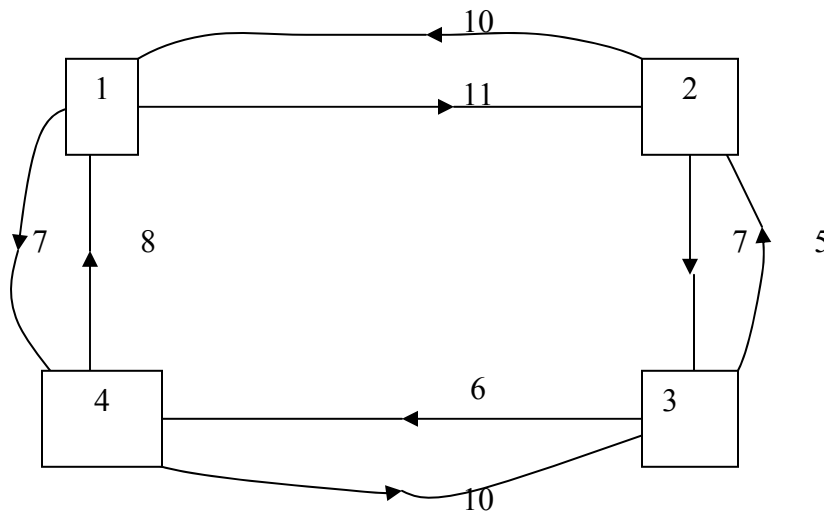
3.5.1 All Or Nothing Assignment Model:

All or nothing assignment is basically an extension of finding the minimum paths through a network. It is called all or nothing because every path from origin zone traffic to a destination zone has either all the traffic (if it is assumed as minimum paths) or none of the traffic.

The steps followed are:

1. Find the minimum path tree from each of the zone centroid nodes to all other nodes.
2. Assign the flow from each origin to each destination node obtained from the trip table to the arcs comprising the minimum path for that movement.
3. Sum the volume on each arc to obtain the total arc volume. If (undirected) link volume is desired, sum the flows on the two arcs that represent bi-directional link.

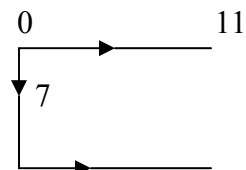
The All or Nothing Traffic Assignment is illustrated using the following example.



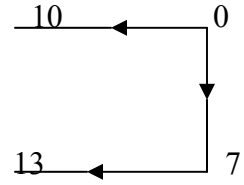
Home Node

1

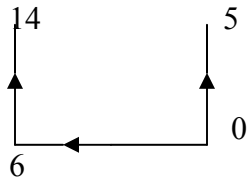
Minimum Path tree



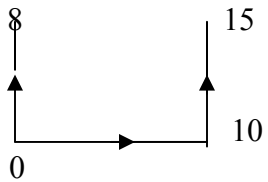
2



3

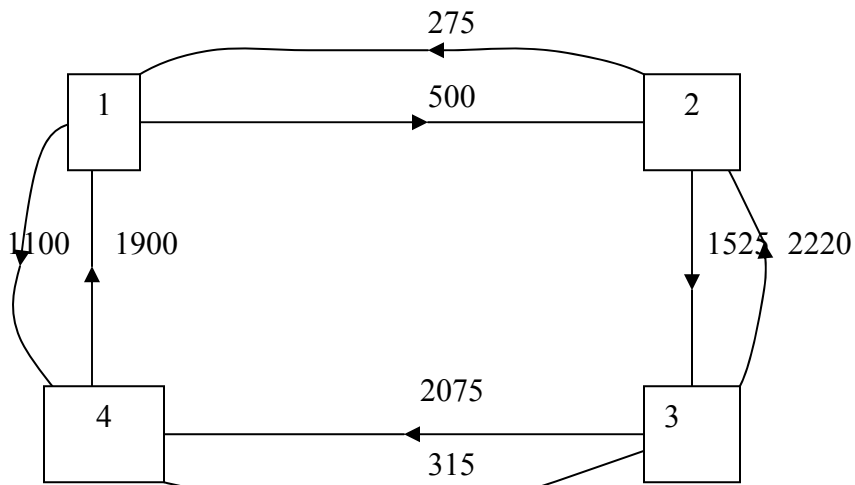


4



Inter node volume, veh/hr is given below

	1	2	3	4
1	-	500	750	350
2	275	-	1050	475
3	650	1870	-	950
4	1250	350	2050	-



LINK 4-3 IS CONGESTED LINK

3.5.2 Capacity Restrained Method:

This model involves two-travel time versus volume relationships used iteratively to arrive at predictions of volumes on up to four separate routes between any two zones. The first equation utilizes route volume as the dependent variable.

The symbolical representation of the graphical portrayal is:

$$t_r = t_{rc} + \frac{d(V_r - V_{rc})L_r}{V_{rc}} \text{-----(1)}$$

where t_r = travel time on route r (minutes)

V_r = Volume of traffic on route r (veh/hr/lane)

V_{rc} = Critical volume for the route r (veh/hr/lane)

t_{rc} = unit travel time at the critical volume (min/mile)

L_r = length of the route r (miles)

d = delay parameter (min/mile)

=0.50 for $V_r < V_{rc}$

=10.0 for $V_r \geq V_{rc}$

The second relationship used for predicting the volume on the route r given travel time:

$$V_r = \frac{1/t_r}{\sum_{r=1}^m 1/t_r} V \text{-----(2)}$$

where V = the total volume of trips from zone I to j on all routes

t_r = travel time on route r (minutes)

V_r = Volume of traffic on route r (veh/hr/lane)

Equation (2) divides up the volume of trips from zone I to j among the various routes in accordance with the reciprocal of the travel times.

The steps which are followed will be:

- i. Find t_r from equation (1)
- ii. Find V_r by inserting t_r in equation (2)
- iii. Enter the value of V_r in equation (1)
- iv. Repeat the same procedure until the changes in volumes or travel times become negligible

Example of the TRC Trip Assignment model

The interplay between two TRC Trip Assignment equations can be demonstrated through a n example in which a pair of zones is connected by the two routes whose characteristics are shown in table 1 below.

Table 1 Example Route Characteristics

Route No.	No. of lanes	Speed limit (mph)	Signals/mile	Length (mile)	Critical volume	Critical travel time	Travel time with no volume
1	1	30	1	3	600	3	2.5
2	1	50	1	4	1100	2	1.5

Iteration 1: - starts with the ideal travel times for the entire length of each route. Thus, if no traffic were:

On route 1, the travel time would equal: 2.5min/mileX3miles = 7.5min

On route 2, the travel time would equal: 1.5min/mileX4miles = 6.0min

This leads to:

$$V_1 = \frac{1/7.5}{1/7.5 + 1/6.0}(1200) = 532vph / lane \text{ and}$$

$$V_2 = \frac{1/6.0}{1/7.5 + 1/6.0}(1200) = 668vph / lane$$

From equation (1) taking $d = 0.5$ since both the routes have volumes less than their respective critical volumes, we get:

$$t_1 = \left[3.0 + \frac{0.50(532 - 600)}{600} \right] 3 = 8.82 \text{ min and}$$

$$t_2 = \left[2.0 + \frac{0.50(668 - 1,100)}{1,100} \right] 4 = 7.20 \text{ min}$$

Iteration 2: -

$$V_1 = \frac{1/8.82}{1/8.82 + 1/7.72} (1200) = 536 \text{ vph/lane and}$$

$$V_2 = \frac{1/7.20}{1/8.82 + 1/7.72} (1200) = 664 \text{ vph/lane}$$

Then the travel times will be:

$$t_1 = \left[3.0 + \frac{0.50(536 - 600)}{600} \right] 3 = 8.85 \text{ min and}$$

$$t_2 = \left[2.0 + \frac{0.50(664 - 1,100)}{1,100} \right] 4 = 7.18 \text{ min}$$

where $d = 0.5$ again since both the routes have volumes less than their respective critical volumes.

Since the last set of travel times do not differ significantly from the previous, the procedure can be terminated and hence the final results are:

$$V_1 = 536 \text{ vph/lane and } V_2 = 664 \text{ vph/lane}$$

$$t_1 = 8.85 \text{ min and } t_2 = 7.18 \text{ vph/lane}$$

The number of iterations is less in this particular case. If the inter zonal volume falls within a range of $\pm 300 \text{ vph/lane}$ of the sum of the critical volumes which in example is $600 + 1,100 = 1,700 \text{ vph/lane}$. The reason for this is that d parameter will jump from 0.5 to 10.0 and back, causing corresponding fluctuations in the related travel times.

3.5.3 Multipath Assignment Model :

McLaughlin developed one of the first multi-path traffic assignment techniques.

A driver route selection criteria is used by McLaughlin which is a function of:

- Travel time
- Travel cost
- Accident potential

The minimum resistance paths between each origin and destination pair are calculated with all the link resistances set to values that correspond to a zero traffic volume. The minimum resistance value between an origin and destination pair is increased by 30%. All the paths between the origin and the destination pair with the resistance values less than this maximum value are identified.

McLaughlin used certain principles of linear graph theory to accomplish the multi-path assignment. Using an electrical analogy it is possible to identify a through variable y that corresponds to current, or traffic flow. A cross variable x may be identified that corresponds to the potential difference, or traffic pressure.

Two postulates from linear graph theory may be introduced that are known as the vertex and circuit postulates.

At any vertex

$$\sum_{i=1}^e a_i y_i = 0$$

where e = the number of oriented terminal graphs, or elements

y_i = the through variable of the i th element

$a_i = 0$ if the i th element is not connected to v

$= 1$ if the i th is oriented away from V

$= -1$ if the i th element is oriented toward

For any circuit,

$$\sum_{i=1}^e b_i x_i = 0$$

where e = the number of oriented terminal graphs, or elements

x_i = the cross variable of the i th element

$b_{ij} = 0$ if the i th element is not in the j th circuit

$= 1$ if the i th element orientation is the same as the j th circuit

$= -1$ if the i th element orientation is opposite to the j th circuit

A sub-graph is then established for each origin and destination pair with these representing two vertices. The connecting elements are the acceptable paths between the vertices plus one flow driver element that corresponds to the car travel demand between the origin and destination pair. The travel demand is assigned among the potential paths building phase. The traffic assigned to each alternative path must be such that the alternative paths have an equal across variable value.

The cross variable X , the resistance value $R(y)$ and the through variable y for each path are assumed to be related as follows:

$$X = R(y)y$$

The above equation is analogous to Ohm's Law in that the potential is equal to the resistance times the flow. In this case the resistance along a path is assumed to be a function of the flow along that path

The figure 1 below represents the way McLaughlin illustrated the assignment method. A schematic two-way street system is shown along with the link descriptions and then trip table. The minimum path trees were determined for all origin-destination pairs using the resistance function for the zero flow, and these are given in table 1 below. Multiplied the minimum path resistance values and the paths whose resistance values were less than this higher value were determined and these paths are also given in *table 2 below*.

The trip table inputs were assigned directly for the origin-destination pairs with only one path. Sub-graphs were formed for the remaining trip table inputs and solved by the chord

formulation of linear graph theory. Figure 2 shows the sub-graph for the origin 1 and destination 3.

The circuit equations may be represented in the general form as:

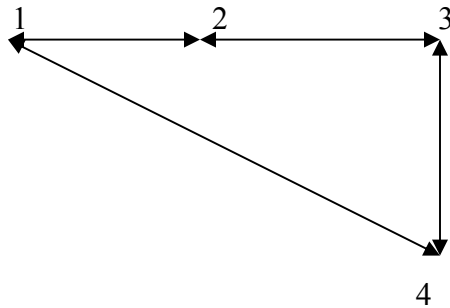
$$\begin{bmatrix} B_{11} & B_{12} & u & 0 \\ B_{21} & B_{22} & 0 & u \end{bmatrix} \begin{bmatrix} X_{b-1} \\ X_{b-2} \\ X_{c-i} \\ X_{c-2} \end{bmatrix}$$

where $B_{11} \dots$ =coefficient matrices corresponding to the branches

u =a unit matrix corresponding to the chords

$X_{b-1} \dots$ =the column matrices of the branches

$X_{c-1} \dots$ = the column matrices of the chords



Link Characteristics

No.	Link	Length (miles)	Maximum Capacity (veh/hr)	Free speed (mph)
1	1-2	1.0	1200	40
2	1-4	2.0	4000	50
3	2-1	1.0	1200	40
4	2-3	0.5	1000	30
5	3-2	0.5	1000	30
6	3-4	0.5	1000	30
7	4-1	2.0	4000	50
8	4-3	0.5	1000	30

Trip Table

Origin	Destination			
	1	2	3	4
1		100	100	2500
2	300*		100	800

Fig 1. Multi-path assignment example problem

Table 1 Paths for multi-path example

Origin	Destination	Minimum path*	Diversion path*	Minimum τ Path	Diversion τ Path
1	2	1,2	-	1,2	-
	3	1,2,3	1,4,3	1,2,3	-
	4	1,4	-	1,4	1,2,3,4
2	1	2,1	-	2,1	-
	3	2,3	-	2,3	-
	4	2,3,4	-	2,3,4	-

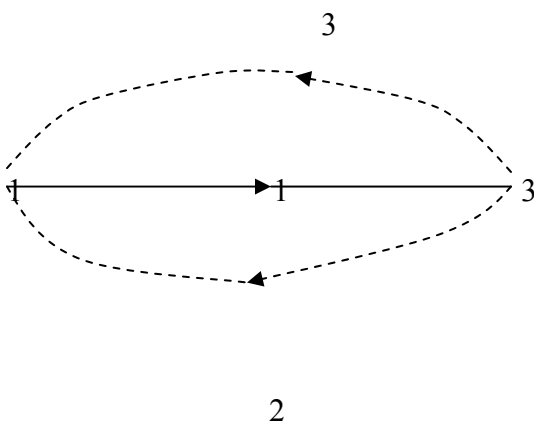
* Based on the resistance function = $s(p)t(p)$

τ based on the resistance function = $S(p)$

Where $s(p)$ = travel cost (operating, accident, and comfort) as a function of the volume -to-capacity ratio p

$t(p)$ = Travel time as a function of the volume-to-capacity ratio p

$S(p)$ = Travel cost (operating, accident, comfort, and time) as a function of the volume -to-capacity ratio p



——— Branch Element 1- path 1,2,3
 - - - - - Chords 2- path 1,4,3
 3-trip table input

Fig 2. Subgraph of the origin 1 and destination 2

The first term is non-existent in this system and the circuit equations are:

$$\begin{bmatrix} B_{12} & u & 0 \\ B_{22} & 0 & u \end{bmatrix} \begin{bmatrix} X_{b-2} \\ X_{c-1} \\ X_{c-2} \end{bmatrix} = 0$$

The terminal equations of the street components may be represented by:

$$\begin{bmatrix} X_{b-2} \\ X_{c-1} \end{bmatrix} = \begin{bmatrix} R_{b-2} & 0 \\ 0 & R_{c-1} \end{bmatrix} \begin{bmatrix} Y_{b-2} \\ Y_{c-1} \end{bmatrix}$$

Where R_{b-2} = the sum of the link resistances corresponding to branch paths

R_{c-1} = The sum of the link resistances corresponding to chord paths

Y_{b-2} = The flow on the branch path

Y_{c-1} = Flow on the chord paths

For the demand assignment:

$$\begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} 9.51 & 0 \\ 0 & 12.17 \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix}$$

The sub-graph fundamental circuit equations are then substituted into the chord set of equations

$$\begin{bmatrix} 0 \\ u \end{bmatrix} X_{c-2} + \begin{bmatrix} B_{12} & u \\ B_{22} & 0 \end{bmatrix} \begin{bmatrix} R_{b-2} & 0 \\ 0 & R_{c-1} \end{bmatrix} \begin{bmatrix} B_{12}^T & B_{22}^T \\ u & 0 \end{bmatrix} \begin{bmatrix} Y_{c-1} \\ Y_{c-2} \end{bmatrix} = 0$$

Where B_{12} = a column matrix with coefficients equal to -1; the number of rows in this matrix correspond to the number of non driver chords in the sub-graph; or it corresponds to the number of paths less 1 between an origin-destination pair

B_{22} = +1, corresponding to the driver or trip table input

Y_{c-1} = The unknown flows for the non driver chord elements

Y_{c-2} = The through driver, or trip table input

The specific formulation for this example is:

$$\begin{bmatrix} 0 \\ 1 \end{bmatrix} X_3 + \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} 9.51 & 0 \\ 0 & 12.17 \end{bmatrix} \begin{bmatrix} -1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} Y_2 \\ 100 \end{bmatrix} = 0$$

Taking the first set of the above equations, the solution is $Y_2=44$ vehicles per hour, and the flow on element 1 is solved by subtraction; $Y_1=100-44=56$ veh/hr. The results of the demand assignment are presented in table 2

Table 2 Link volumes for multi-path example

No.	Link	R_0	Y_1	R_1	y_2	y_3	y_4
1	12	4.90	156	4.98	138	1,290	625
2	14	7.56	2,544	8.42	2,562	1,410	2,075
3	21	4.90	300	5.00	300	300	300
4	23	4.61	956	16.40	938	2,090	1,425
5	32	4.61	0	4.61	0	0	0
6	34	4.61	800	5.80	1,890	1,890	1,225
7	41	7.56	0	7.56	0	0	0
8	43	4.61	44	4.61	62	0	0

In McLaughlin’s assignment procedure, new link and path resistance values are calculated for the capacity restraint assignment that corresponds to the flows obtained from the demand assignment. The procedure described above is employed again to calculate the restrained volumes. If these volumes are within tolerable limits of the demand volume then the restraint assignment is complete, otherwise an iterative is required.

An iterative solution is achieved by averaging the link volumes according to the following expression:

$$\bar{Y} = \sum_{i=1}^n \frac{y_i}{n}$$

Where \bar{Y} = the average assigned volumes

Y_i = The trips assigned to the links during the *ith* iteration of the linear graph procedure including iterations

n = the number of linear graph iterations

Burrell has proposed a technique for generating multiple paths through a traffic network. This method assumes that the user does not know the actual travel times on the links but associates a supposed travel time on each link that is drawn at random from a distribution of times. It assumes that the user finds and uses a route that minimizes the sum of the supposed link times.

Burrell assumes that a group of trips originating from a particular zone have the same set of supposed link times and consequently there is only one tree for each zone of production. A rectangular distribution that could assume eight separate magnitudes was assumed and the ranges of distributions for each of the links were selected so that the ratio of the mean absolute deviation to actual link time was the same for all links. The demand or capacity restrained assignments are then made to the paths selected in the above manner.

Another multi-path assignment technique has been proposed by Dial. With this technique each potential path between a particular origin and destination pair is assigned a probability of use that then allows the path flows to be estimated.

3.6 GRAPH THEORY APPROACH

Graph theory is basically a branch of topology. Geometric structure of a transport network, which is the topological pattern formed by nodes and routes is studied by graph theory. The use of graph theoretic approach in road network analysis is to compare and evaluate one network analysis with the other which may be in the same region or in different regions. It can be also used to check connectivity and accessibility level of different nodes.

Connectivity:

Its concept involves following terms:

1. **Degree of vertex:** Number of edges meeting at the vertex.
2. **Path:** Collection of vertices and a subset of their incident edges so that degree of each internal vertex is two or more and the degree of each terminal vertex is one.
3. **Circuits:** Closed path where all vertices are of degree two or more.
4. **Connected and Unconnected Graph:** Connected if there exist at least one path between any pair of vertices in graph. In unconnected, there are pairs of points or vertices which cannot be joined by a path.

Structural and geometrical properties of alternative transport networks can be measured in terms of following graph theoretic measures:

1. **Beta Index:** Ratio of total number of links to the total number of nodes in network.

Mathematically: $\beta = (e/v)$

Where: e and v are, respectively, number of edges and vertices in network.

2. **Cyclomatic Number:** A count of the number of fundamental circuits existing in the graph. It is an measure of redundancy in the system.

Mathematically: $\mu = e - (v - p)$

Where: p is number of maximal connected sub graph.

3. **Gamma Index:** Ratio of the observed number of edges in network to maximum number of edges which may exist between specified number of vertices.

Mathematically: $\gamma = e \times 100 / (3(v - 2))$

4. **Alpha Index:** Ratio of the observed number of fundamental circuits to the maximum possible number of complete circuits.

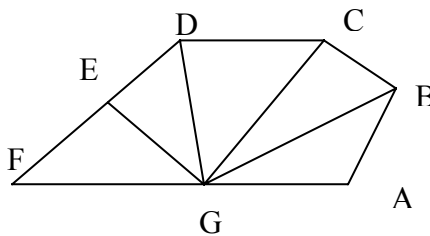
Mathematically: $\alpha = \mu \times 100 / (2v - 5)$

5. **Associate Number:** The number of links needed to connect a node to the most distant node from it. The node which has low associate number is most accessible.
6. **Shimbel Index:** Measure of accessibility which indicates the number of links needed to connect any node with all other nodes in the network by the shortest path. The node having lower shimbel index is the most accessible.
7. **Dispersion Index:** It is the measure of connectivity of transport network and obtained by sum of the shimbel index.

$$\text{Mathematically, } DI = \sum_{j=1}^v \sum_{i=1}^v d_{ij}$$

8. **Degree of Connectivity:** Ratio of maximum possible number of routes to have
 Complete connectivity to observed number of routes in network.
 Degree of Connectivity = $((v(v - 1)/2)/e$

Example 1:



The solution for the problem can be done in tabular form as follows:

Table giving Solution for above figure

	A	B	C	D	E	F	G	A.N.	S.I
A	-	1	2	2	2	1	2	2	10
B	1	-	1	2	2	1	2	2	9
C	2	1	-	1	2	1	2	2	9
D	2	2	1	-	1	1	2	2	9
E	2	2	2	1	-	1	1	2	9
F	1	1	1	1	1	-	1	1	6
G	2	2	2	2	1	1	-	2	10
							Total	13	62

Mean Associate Number/Vertex = $13/7 = 1.8571$

Mean Dispersion Index/Vertex = $62/7 = 8.85$

As mean associate number and mean dispersion index for this network is less. So from accessibility point of view, this network is better.

Node link incidence matrix:

It is an $n \times l$ matrix E whose element in the row corresponding to node i and the column corresponding to the link (j, k) is defined to be:

$$\begin{aligned}
 &+1 && \text{if } i = j, \\
 &-1 && \text{if } i = k, \text{ and} \\
 &0 && \text{otherwise.}
 \end{aligned}$$

By multiplying node link incidence matrix (E) with chain flow vector (f), we get $O - D$ flow vector. So mathematically,

$$E \times f = g$$

Where:

E is the $n \times l$ node link incidence matrix,

f is the $l \times 1$ link flow vector, and

g is the $n \times 1$ $O - D$ flow vector.

3.7 Flows and Conservation Law (Kirchhoff's Law)

Fundamental to the theory of flow of electric currents in electrical networks, water in pipe networks, or traffic in transportation networks, is Kirchhoff's law, which is a conservation law stating that, for steady state conditions imply for traffic applications that we are not concerned with the microscopic and stochastic characteristics of a traffic stream of individual vehicles traveling at random or in platoon on a city street network, but rather with the gross macroscopic behavior of traffic as, for example, on a main road network. We ignore fluctuations over time.

Kirchhoff's law states that "the sum of all flows leaving an intermediate node equals the sum of all flows entering the node".

Kirchhoff's law then states that "the sum of all flows leaving the centroid equals the flow produced at the centroid, and the sum of all flows entering the centroid equals the flow attracted to the centroid."

We shall adopt for general transportation network the terminology of centroids and intermediate nodes to distinguish between nodes where traffic may be, and may not be, produced or attracted. In many other applications, the centroids are called source and sinks. We shall adopt the following notation. The link flow on the directed link (i, j) will be denoted by f_{ij} , the flow produced at a centroid i by a_i , and the flow attracted i by b_i . The quantities f_{ij} , a_i , b_j are assumed to be nonnegative. It is convenient to define $A(i)$ and $B(i)$, the set of nodes "after" and "before" node i by

$$A(i) = \{j/j \in N, (i, j) \in CL\}, \quad (1)$$

$$B(i) = \{j/j \in N, (j, i) \in CL\}. \quad (2)$$

Kirchhoff's law for a directed transportation network $[N, L]$ can then be written in the form of conservation equations as follows:

If i is a centroid then the following formulae should be satisfied

$$\sum_{A(i)} f_{ij} = a_i \quad (3)$$

$$\sum_{B(i)} f_{ji} = b_i \quad (4)$$

if I is an intermediate node then the following formulae should be satisfied

$$\sum_{A(i)} f_{ij} - \sum_{B(i)} f_{ji} = 0, \quad (5)$$

For these equations to have solutions, the total production, $\sum a_i = r$ say, must be equal to the total attraction $\sum b_i$. Since the number of links is generally at least twice the number of node in a network, the number of unknowns in Eqs. (3)- (5) greatly exceeds the number of equations and the equations are rich in solutions.

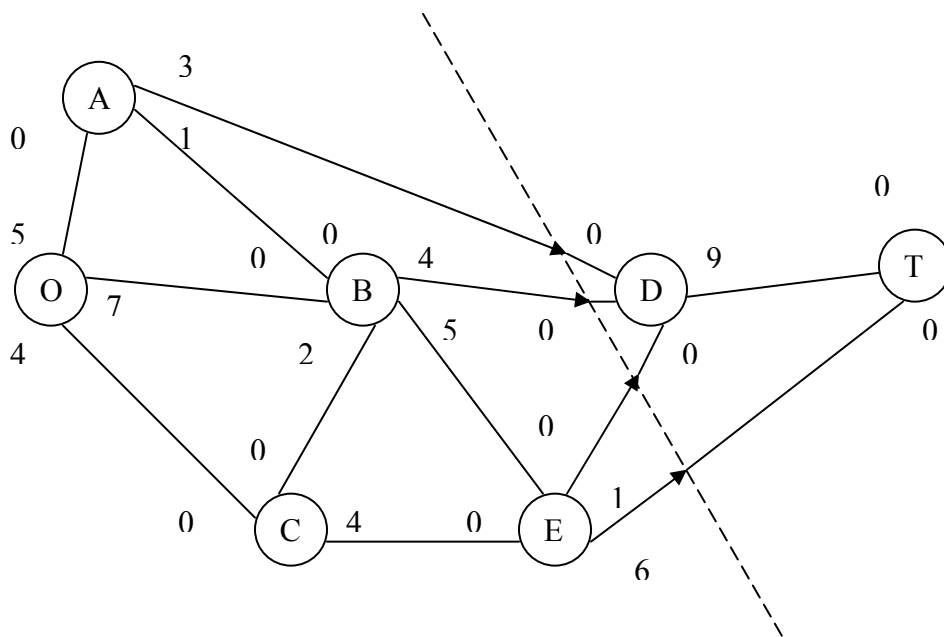
Figure 1 illustrates a transportation network with two centroids and two intermediate nodes. For the intermediate node 2, Kirchhoff's law can be easily verified:

$$i=2, \quad A(2) = \{3,4\}, \quad B(2) = \{1,3\},$$

$$\sum_{A(2)} f_{2j} - \sum_{B(2)} f_{j2} = f_{23} + f_{24} - f_{12} - f_{32} = 5 + 1 - 3 - 3 = 0$$

3.8 Min Cut Max Flow Theorem:

In max flow min cut theory, a cut may be defined as any set of directed arcs containing at least one arc from every directed path from source to sink. There normally are many ways to slice through a network to form a cut to help analyze the network. For any particular cut, the cut value is the sum of the arc capacities of the arcs (in the specified direction) of the cut. The **max-flow min-cut theorem** states that, for any network with single source and sink, the maximum feasible flow from the source to the sink equals the minimum cut value for all cuts of the network.



Considering above network, one interesting cut through this network is shown by dotted line. Notice that the value of the cut is $3 + 4 + 1 + 6 = 14$, which was found to be the maximum flow value. So this cut is a minimum cut.

3.9 Dynamic Stochastic Assignment Model with Many Routes between a Single O-D Pair

Let D be the demand for travel from the origin zone h to the destination zone d and assume that the period between the earliest and latest possible times departure from h is considered as a series of equal spaced epochs, representing intervals of length w .

Suppose the available routes are numbered $1, 2, \dots, r, \dots, I$ and that $D_r(k, j)$ is the number of vehicles for unit time departing from h , on route r at epoch k on day z .

Let $X_a(k, z)$ and l_a denote the number of vehicles on any link, a , at epoch k on day z and the link length respectively.

The number of vehicles leaving link a per unit time at epoch k on day z is

$$V_a(k, z) = X_a(k, z) * S_a(k, z) / l_a$$

The flow conservation equations for link a may be written as

$$X_a(k+1, z) = X_a(k, z) + w \left\{ \sum_1^z D_r(k, z) \delta_{ar} - V_a(k, z) \right\} \text{ For } a \in I_h$$

Where $\delta_{ar} = 1$ if link a is on route r
 0 otherwise

$$X_a(k+1, z) = X_a(k, z) + w \left\{ \sum_m P_{am}(k, z) V_m(k, z) - V_a(k, z) \right\} \text{ if } a \text{ isn't on the route } h$$

where M_a is the set of links having as final node the initial node of link a , and $P_{am}(k, z)$ is the proportion of the vehicles leaving link m at epoch k on day z which enter link a .

In a test with a small network Vythoulkas was not able to contain a stable situation; the departure pattern tended to oscillate about a possible equilibrium distribution.

3.10 Dial's Model:

Dial's model is based on the hypothesis that there should be a non-zero probability of use of all 'efficient' paths. He proposed two definitions of efficient path, namely:

1. a path in which every link has its initial node closer than its final node to the origin and has its final node closer than its initial node to the destination; and
2. a path in which every link has its initial node closer than its final node to the origin.

Conceptually the first of these definitions is more attractive but it has the drawback that in any single run trees can be built from origin zone to one destination zone only because, for each node in the network, the minimum cost to each destination zone must be known as well as that from the origin zone under consideration. The second definition lends itself to the usual 'once through' approach.

Basically, trips are allocated to any efficient route R so that,

$$V_R = V^* \exp(-\theta(c_R - c^*)) = V^* \exp(-\theta\Delta_R) \text{-----(1)}$$

Where $\Delta_R = c_R - c^*$ is the excess cost in using route R rather than the minimum cost route, the cost on which is c^* , and V_R and V^* are the flows on route R and the minimum cost route respectively.

The value of θ determines the proportions of the trips allocated to the efficient paths; if $\theta = 0$, then the trips are shared equally between them but a high value of θ produces a heavy bias towards the cheapest routes.

Dial's route (vine) building and link loading algorithm

The following algorithm may be used to simultaneously assign trips from the origin node to all destination nodes in accordance with Dial's second definition of the efficient paths:

Step 1:

Determine the minimum path costs from the origin node h to all nodes in the network. Let c_j^* denote the minimum path cost from h to node j and let d_m be the destination node with greatest minimum path cost from h.

Step 2:

i. Initialize all nodal weights:

Set all node weights $w_j=0$; set origin node weight $w_h = 1$.

ii. Consider each node j in the network, in order of increasing minimum path cost, c_j^* , until d_m is reached, as indicated below.

a. For each link (i,j) with final node j:

Calculate the ‘effectiveness’

$$e_{ij} = \begin{cases} \exp(-\theta\Delta_{ij}) & \text{if } c_i^* < c_j^* \\ 0 & \text{otherwise} \end{cases} \text{-----}(2)$$

where c_{ij} is the cost on link (i, j), and $\Delta_{ij} = (c_i^* + c_{ij}^*) - c_j^*$ is the excess cost involved in going from origin node h to node j via link (I, j) rather than by the minimum cost path; and

Calculate its link weight $w_{ij} = w_i e_{ij}$ -----(3)

b. Determine the node weight $w_j = \sum_i w_{ij}$ -----(4)

(the ratio $\frac{w_{ij}}{\sum_i w_{ij}} = \frac{w_{ij}}{w_j}$ is effectively the probability of a trip from origin h

arriving at node j via the predecessor link (i, j), rather than via the other links with final node j)

Step 3: (Backward pass)

a. Initialize all nodal volumes

Set $v_j = 0$ for all nodes j ; and

Set $v_d = T_{h-d}$, the trips from h to d , for all destination nodes d .

b. Starting with destination node d_m , consider each node j in the network in the order of decreasing c_j^* as indicated below.

For each link (i, j) with final node j :

i. Derive the link volume $v_{ij} = v_j \frac{W_{ij}}{W_j}$; and

ii. Set $v_i = v_i + v_{ij}$

Terminate when $j =$ the origin node, h .

Note: - if P is any path from the origin node h to destination node d , via nodes i, j, k, l, \dots, y, z , say, and Δ_p is the excess cost on the path, then:

$$\begin{aligned} \Delta_p &= (c_{hi} + c_{ij} + c_{jk} + c_{kl} + \dots + c_{yz} + c_{zd}) - c_d^* \\ &= -c_h^* + ((c_h^* + c_{hi}) - c_i^*) + ((c_i^* + c_{ij}) - c_j^*) + ((c_j^* + c_{jk}) - c_k^*) \\ &\quad + \dots + ((c_z^* + c_{zd}) - c_d^*) \\ &= \Delta_{hi} + \Delta_{ij} + \Delta_{jk} + \dots + \Delta_{zd}, \text{ Since } c_h^* = 0 \end{aligned}$$

Following step 2 above, the probability of choosing path P from h to d is:

$$\begin{aligned} P_r(p) &= \frac{W_{zd}}{W_d} \cdot \frac{W_{yz}}{W_z} \cdot \frac{W_{xy}}{W_y} \cdot \dots \cdot \frac{W_{ij}}{W_j} \cdot \frac{W_{hi}}{W_i} \\ &= \frac{W_z \cdot e_{zd}}{W_d} \cdot \frac{W_y \cdot e_{yz}}{W_z} \cdot \frac{W_x \cdot e_{xy}}{W_y} \cdot \dots \cdot \frac{W_i \cdot e_{ij}}{W_j} \cdot \frac{W_h \cdot e_{hi}}{W_i} \\ &= \frac{1}{W_d} \exp(-\theta(\Delta_{zd} + \Delta_{yz} + \Delta_{xy} + \dots + \Delta_{ij} + \Delta_{hi})) \text{ Since } W_h = 1. \end{aligned}$$

$$= \frac{1}{W_d} \exp(-\theta(\Delta_p))$$

$$= \frac{E_p}{W_d} \text{ where } E_p = \exp(-\theta(\Delta_p)) \text{ is the "effectiveness" of path p} \text{-----(5)}$$

If S is the set of all "efficient" paths from h to d, then $\sum_{P \in S} P_r(p) = 1$

Therefore, $\sum_{P \in S} \frac{E_p}{W_d} = 1$ and $W_d = \sum_{P \in S} E_p$

Since, for the minimum cost path, p^* , say, $E_{p^*} = 1$, w_d may be regarded as the effective number of 'efficient' paths from h to d.

Example 1:

In the network shown in figure 1 below, in which all the links are on one way, using Dial's method assign 4,000, 2,000 and 1, 000 trips from node 1 to nodes 6, 8 and 9 respectively. Assume $\theta = 1$.

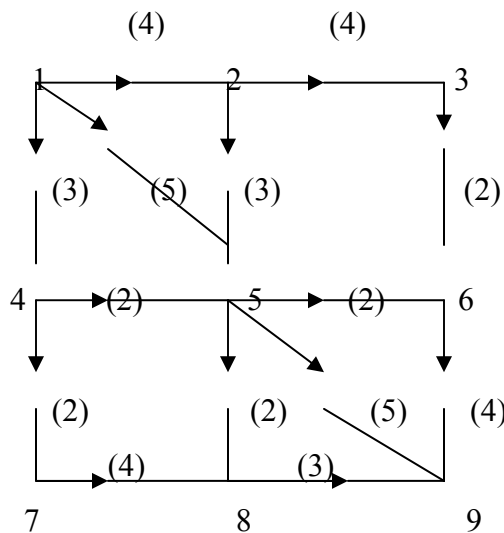


Figure.1.1 Simple network for the Example

Table: Network Table

Link No	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Initial Node	1	1	1	2	2	3	4	4	5	5	5	6	7	8
Final Node	2	4	5	3	5	6	5	7	6	8	9	9	8	9
Cost	4	3	5	4	3	2	2	2	2	2	5	4	4	3

Step 1: The minimum path costs from origin 1 to all other nodes can be determined by inspection. In a normal-sized network they could be determined by any of the tree building algorithms.

Table: Nodal Weights

Node j	W_j	
	Initial	Final*
1	1.0	
2	0	1.0000
3	0	1.0000
4	0	1.0000
5	0	2.1353
6	0	2.1353
7	0	1.0000
8	0	2.2706
9	0	5.1915

* Final $W_j = \sum_i W_{ij}$

Table: Link Weights

Node j	c_j^*	Links in $F_j(a)$	Initial Node of link in $F_{j,i}$	e_{ij}	w_{ij}
1	0				
4	3	2	1	1.0000	1.0000
2	4	1	1	1.0000	1.0000
5	5	3	1	1.0000	1.0000
		5	2	0.1353	0.1353
		7	4	1.0000	1.0000
7	5	8	4	1.0000	1.0000
6	7	6	3	0	0(b)
		9	5	1.0000	2.1353
8	7	10	5	1.0000	2.1353
		13	7	0.1353	0.1353
3	8	4	2	1.0000	1.0000
9	10	11	5	1.0000	2.1353
		12	6	0.3679	0.7856
		14	8	1.0000	2.2706

Note: a. F_j is the set of all links with final node j

b. Link no.6 is not 'efficient' since $c_3^* < c_6^*$. It may be ignored hereafter

Table: Nodal Volumes

Node j	V_j
1	7000.2 τ
2	434.3
3	
4	145.2
5	6854.8
6	151.3
7	145.2
8	2437.4
9	1000

τ Differs from 7000 due to rounding

Table: Link Volumes

Node j	c_j	Links in Fj	Initial Node link in $F_{j,i}$	w_j	w_{ij}	$v_{ij}\tau$
9	10	11	5	5	2.1353	411.3
		12	6	1915	0.7856	151.3
		14	8		2.2706	437.4
3	8	4	2	1.0000	1.0000	0
8	7	10	5	2.2706	2.1353	2292.2
		13	7		0.1353	145.2

6	7	9	5	2.1353	2.1353	4151.3
7	5	8	4	1.0000	1.0000	145.2
5	5	3	1	2.1353	1.0000	3210.2
		5	2		0.1353	434.3
		7	4		1.0000	3210.2
2	4	1	1	1.0000	1.0000	434.3
4	3	2	1	1.0000	1.0000	3355.7
1	0			1.0000		

τ The link loads in the final column relate to the links indicated in the third column

Selected link analysis for Dial’s assignment procedure (second definition of ‘efficient’ path)

Although the ‘efficient’ paths between individual origin-destination node pairs, h-d, are not explicitly defined when Dial’s assignment procedure is used, it is possible to determine the volume on any link (i, j) arising from the trips h to d using an algorithm proposed by Van Vliet (1981)

If $c_i^* > c_j^*$, where c_i^* and c_j^* are the minimum path costs from the origin node h to nodes I and j respectively, then the link (i, j) is excluded from the efficient paths from h and hence there are no trips from the origin node h on link (i, j).

Assuming $c_i^* < c_j^*$, we can proceed as follows:

Let X_i denote the set of all ‘efficient’ paths, p, from origin node h to node i and let Y_j denote the set of all ‘efficient’ paths, Q, from node j to the destination node d.

Let $W_{h-i} = \sum_{p \in X_i} E_p$ and $W_{j-d} = \sum_{Q \in Y_j} E_Q$, where E_p and E_Q are already defined, but in relation to paths from h to i and from j to d respectively.

Van Vliet has shown that if trips have been assigned from origin node h to destination node d in accordance with Dial's algorithm, then the probability of a trip from h to d passing along link (i, j) is given by:

$$P_r((i, j)/(h-d)) = e_{ij} W_{h-i} W_{j-d} / W_d \text{-----(6)}$$

If there are T_{h-d} trips to be assigned from origin node h to destination node d, then the number of these trips passing along link (i, j) is $T_{h-d} P_r((i, j)/(h-d))$.

On the basis of this relationship, the loading on link (i, j) arising from the assignment of the trips T_{h-d} may be determined as follows: -

Step1: (A standard forward pass)

- a. Carry out steps 1 and 2 as for the standard Dial algorithm.

In course of step 2: e_{ij} is determined; W_{h-i} is determined as W_i ; and the W_d are determined.

- b. Set $W'_d = W_d$ for all destination nodes d.

Step 2: (Modified forward pass to determine W_{j-d})

- a. Repeat step 2(a) of the standard Dial algorithm but with $W_j = 1$, not $W_h = 1$.
- b. Calculate link weights W_{ij} and node weights W_j as steps 2(b)(i) and (ii) of the standard Dial algorithm but when considering each node in the network in order of increasing cost from the origin node h, consider only those nodes coming after node j.

(The values of W_d obtained from step 2 are the required values of W_{j-d} , i.e, set $W_{j-d} = W_d$ from the table of nodal weights at the end of step 2).

Step 3: (Determination of link volumes)

- a. Substitute the values of e_{ij}, W_{h-i} and $W_d = W'_d$ from step 1 and W_{j-d} from step in equation (?) to obtain, for each h-d pair, the value of $P_r((i, j)/(h - d))$.
- b. Calculate $V_{ij}(h - d) = T_{h-d} P_r((i, j)/(h - d))$ and output results.

Note: Van Vliet (1981) actually considered the select link analysis for a set of links, i.e, link (i,j) in the above is replaced by a sub-route from i to j.

Example 2: Determine the loading on link (4,5) in Example 1.

We have: origin node, $h = 1; d_j = 6,8,9; T_{1-6} = 4000; T_{1-8} = 2000; T_{1-9} = 1000$

Step 1: From the previous example of table 2, $W_{h-i} = W_{1-4} = W_4 = 1.0000$; for the destination nodes- $W'_6 = 2.1353; W'_8 = 2.2706$; and $W'_9 = 5.1915$. From table 3, $e_{45} = 1.0000$

Step 2: (Modified forward pass)

Table: Nodal Weights

Node s	$\tau W_s (= W_{5-s})$	
	Initial	Final
1	0	
2	0	
3	0	
4	0	

5	1.0000	
6	0	1.0000
7	0	
8	0	1.0000
9	0	2.3679

τ Final $W_s (= W_{5-s}) = \sum_r W_{rs}$ from table 1.6 below.

Table: Link Weights

Node s	c_s^*	Links in F_s	Initial Node of link in $F_{s,r}$	e_{rs}	W_{rs}
7	5	8	4	1.0000	0
6	7	6	3	0	0
		9	5	1.0000	1.0000
8	7	10	5	1.0000	1.0000
		13	7	0.1353	0
3	8	4	2	1.0000	0
9	10	11	5	1.0000	1.0000
		12	6	0.3679	0.3679
		14	8	1.0000	1.0000

Table: Select Link Loading for Link (4,5)

i	j	e_{ij}	W_{1-i}	d	τW_{j-d}	W'_d	$P_r((i, j)/(1-d))$	T_{1-d}	$V_{ij}(1-d)$
4	5	1.0000	1.0000	6	1	2.1353	0.4683	4000	1873.3
				8	1	2.2706	0.4404	2000	880.8
				9	2.3679	5.1915	0.4561	1000	456.1

$\tau W_{j-d} = W_d$ from the table 1.6

Table: Output

Load on Link (4,5)	Origin Node	Destination Node
1873.3	1	6
880.8	1	8
456.1	1	9
Total	3210.2	

3.11 Gunnarsson's Model:

Gunnarsson (1972) said of his model that it is 'seemingly close in intent' to Dial's model 'but vastly different in both strategy and tactics'. In concept, the method differs from that of Dial in that whereas Dial considered the entire routes from origin to destination when making a choice, Gunnarsson assumed that a driver makes a choice of route at each node arrived at, independently of any previous decision. He considered a reasonable route from origin node to destination node d to be one such that every link in it has its final node nearer to d than its initial node and is such that the cost on the route is within chosen limits of the cost on the minimum cost path.

He assumed that at any node, i , the basis of the choice of the next link, (i, j) say, in the path to the destination node d , is the cost from i to d via link (i, j) , which he referred to as the resistance of link (i, j) , denoted by r_{ij} . Letting d_j^* and c_{ij} denote the minimum cost to d from any node j and the cost on link (i, j) respectively.

$$r_{ij} = c_{ij} + d_j^* \text{-----}(7)$$

The probability of use of link (i, j) is then given by:

$$P_r(i, j) = f(i, j) / \sum_{(i,k) \in I_i} f(r_{ik}) \text{-----}(8)$$

where $f(r_{ij})$ is some function of r_{ij} and I_i is the set of all links with initial node i .

Following traffic studies, he used $f(r_{ij}) = r_{ij}^{-8}$. To prevent excessively costly routes being used, he introduced an acceptable 'prolongation' factor, w , for a route. To be a part of a reasonable route, a link (i, j) should satisfy the following conditions.

1. $d_j^* < d_i^*$; and
2. $r_{ij} \leq wd_i^*$

A vine building and link loading algorithm for Gunnarsson's model

The following is an algorithm, for simultaneously assigning trips from all origin nodes to destination node d , in accordance with his definition of a reasonable path: -

(It is assumed that the value of the acceptable prolongation factor, w , and the form of the probability function $f(r_{ij})$ have been specified).

Step1: (Determination of d_i^*)

Set $c'_{ij} = c_{ji}$ for all links (j, i) .

Using the link cost matrix $\{c'_{ij}\}$ determine the minimum path costs, d_i^* , from d to all nodes, i . Let h_m be the origin node with greatest minimum path cost to d .

Step 2: (Determination of link and nodal attractivities A_{ij} and a_i .)

Starting with h_m , consider each node i in the network in order of decreasing d_i^* until d is reached.

For each node i :

a. For each link (i,j) with initial node i , i.e, for all $(i,j) \in I_i$,

i. determine its resistance $r_{ij} = c_{ij} + d_j^* (= c_{ji}^* + d_j^*)$, and

ii. determine its attractivity (weight) A_{ij} where

$$A_{ij} = f(r_{ij}) \text{ if } d_j^* < d_i^* \text{ and } r_{ij} < wd_i^*, \text{ otherwise } A_{ij} = 0;$$

b. Determine the total attractivity of node i , using $a_i = \sum_{(i,k) \in I_i} A_{ik}$

Step 3: (Determination of nodal volumes V_j and link volumes V_{ij})

a. Initialize all nodal volumes:

Set $V_i = 0$ for all nodes i ; and

Set $V_h = T_{h-d}$, the trips from h to d , for all origin nodes h .

b. Starting with origin node h_m , for each node i in order of decreasing d_i^* :

- i. for each link (i,j) in I_i , derive the link volume $V_{ij} = V_i A_{ij} / a_i$; and
- ii. set $V_j = V_j + V_{ij}$

Stop when node d is the node considered.

Gunnarsson's model may be brought closer to Dial's by specifying:

$$f(r_{ij}) = \exp(-\theta(r_{ij} - d_i^*)) = \exp(-\theta \nabla_{ij}),$$

where $\nabla_{ij} = (c_{ij} + d_j^*) - d_i^*$ is the excess cost involved in proceeding to destination node from node i via link (i,j) rather than by the minimum cost path from i. With this form for $f(r_{ij})$, the probability that, of all the links available at node i, link (i,j) is chosen for the continuation of the journey to d is given by:

$$P_r(i, j) = \frac{f(r_{ij})}{\sum_{(i,k) \in I_i} f(r_{ik})} = \frac{\exp(-\theta \nabla_{ij})}{\sum_{(i,k) \in I_i} \exp(-\theta \nabla_{ik})} = \frac{A_{ij}}{a_i} \text{-----(9)}$$

Example1:

In the network in Example 1, reverse all a link directions and assign 4000, 2000 and 1000 trips to node 1 from nodes 6, 8 and 9 respectively using Gunnarsson's model with $W = 1$ and $f(r_{ij}) = \exp(-\nabla_{ij})$.

Step 1: The minimum path costs to node 1 from all other nodes are the same as those from node 1 to the other nodes using the un-reversed link cost.

Step 2:

Table: Link and Nodal Attractivities A_{ij}, a_i

Node i	d_i^*	Final Node of link in $I_{i,j}$	c_{ij}	d_j	r_{ij}	∇_{ij}	A_{ij}	a_i
9	10	5	5	5	10	0	1.0000	2.3679
		6	4	7	11	1	0.3679	
		8	3	7	10	0	1.0000	
3	8	2	4	4	8	0	1.0000	1.0000
8	7	5	2	5	7	0	1.0000	1.1353
		7	4	5	9	2	0.1353	
6	7	3	2	8			0	1.0000
		5	2	5	7	0	1.0000	
5	5	1	5	0	5	0	1.0000	2.1353
		2	3	4	7	2	0.1353	
		4	2	3	5	0	1.0000	
7	5	4	2	3	5	0	1.0000	1.0000
2	4	1	4	0	4	0	1.0000	1.0000
4	3	1	3	0	3	0	1.0000	1.0000
1								0

Table: Nodal Volumes

Node j	V_j
1	7000.0
2	425.3
3	
4	3431.7
5	6711.3
6	4155.4
7	288.7
8	2422.3
9	1000

Table : Link Volumes

Node i	d_i^*	Final Node of link in $I_{i,j}$	A_{ij}	a_i	V_{ij}
9	10	5	1.0000	2.3679	422.3
		6	0.3679		155.4
		8	1.0000		422.3
3	8	2	1.0000	1.0000	0
8	7	5	1.0000	1.1353	2133.6
		7	0.1353		288.7
6	7	5	1.0000	1.0000	4155.4
5	5	1	1.0000	2.1353	3143.0
		2	0.1353		425.9
		4	1.0000		3143.0
7	5	4	1.0000	1.0000	288.7
2	4	1	1.0000	1.0000	425.3
4	3	1	1.0000	1.0000	3431.7

Selected Link Analysis for Gunnarsson's Assignment Procedure

The procedure proposed below for determining the trips from the various origin zones h which include link (i,j) in their paths to destination node d , is an adaptation of the method for carrying out a selected link analysis with GMTU assignment procedure.

The procedure for determining the movements to destination node d which include link (i, j) in their paths is as follows: -

Step 1: (Determination of the minimum path costs, d_i^* , from all nodes, i , to the destination node d).

Step 2: (Determination of the link attractives)

- a. Repeat step 2 of the algorithm but terminate when node i has been processed. Results in the link attractives matrix (LA) are, for each node m for which $d_m^* \geq d_i^*$ in decreasing order of magnitude of d_m^* : the node number m ; a list of pairs of values of successor node n (the final node of a link with initial node m) and A_{mn} ; and $a_m = \sum_{(m,n) \in I_m} A_{mn}$
- b. Check if j is a successor node of I ; if it is not then no minimum paths to d pass along (i,j) ; if it is store $P_r(i, j) = A_{ij} / a_i$

Step 3: (Setting up the pointer array)

Starting at the top of array LA, i.e., with node h_m , and proceeding down the rows for each row:

- a. read node number, m , the list of pairs of successor nodes n and corresponding A_{mn} , and a_m .
- b. if for any successor node, n , $d_n^* < d_i^*$, delete n and A_{mn} ;
- c. replace all remaining values of A_{mn} by $P_r(m, n) = A_{mn} / a_m$;
- d. for each successor node n remaining in the list, enter m and $P_r(m, n)$ in row n of the pointer array.

Step 4: (Setting up an array N, consisting of nodes n which are such that at least one path from node n to d passes through node i)

- a. Set up array X with elements $x_r = 0$ for $r=1$ to N_{nodes} .
- b. Enter I in array N and set $x_i = 1$
- c. Read the next entry, n, in N (n=i initially); stop if there are no more nodes in N.
- d. Read the entries in row n of the pointer array. If there are no entries, go to c, otherwise, for each node m entered:
If $x_m = 0$, set $x_m = 1$ and enter m in array N; if $x_m \neq 0$ do nothing.
- e. Go to (c)

Step 5: (Setting up an 'ordered' pointer array)

Arrange the nodes, n in N in order of increasing d_n^* and enter nodes n together with the entries in row n of the pointer array in an 'ordered' pointer array.

Step 6: (Setting up a node weighing array, W, containing for each node n in N, the probability that a path from n to d passes along link (i,j)).

- a. Initialize array W
Set $W_i = P_r(i, j)$; set $W_n = 0$ for all other nodes n in N.
- b. From each row of the 'ordered' pointer array in turn:
 - i. read the node pointed to, n, (equal to i to start with), and the list of pairs of predecessor nodes m and probabilities $P_r(m, n)$;
 - ii. for each predecessor node m, calculate $W_m = W_n \cdot P_r(m, n)$ and set $W_m = w_m + W_m$.

Step 7: (For all origin nodes h, calculation of the trips to destination node d from h passing along (i,j) and output of the link loading information).

- a. From the trip matrix, read off the origin nodes, h, and T_{h-d} , the trips from h to d.

- b. Consider each origin node, h, in turn. If h is not in array N select the next origin node, if any, else if h is in N: -
 - i. Calculate $V_{ij}(h-d) = W_h \cdot T_{h-d}$ and,
 - ii. Output a message indicating that there are $V_{ij}(h-d)$ trips on link (i,j) from origin node h to destination node d.

Example 2:

Determine the loading on link (4,1) in Example 3 above.

Step 1: The minimum path costs to destination node d=1 from all other nodes are the same as those from node 1 to the other nodes in Example 1.

Step2: The table link attractivities, LA, is the same as table 2.1 for the nodes down to and including node i=4. j=1 is a successor node of i=4; $P_r(4,1) = A_{41} / a_4 = 1.0000$.

Step3:

Table 1.2.4 Pointer Array

Node (row), n	Predecessor nodes, m, and corresponding $P_r(m,n)$		
1			
2	3(1.0000)	5(0.0634)	
3			
4	5(0.4683)	7(1.0000)	
5	9(0.4223)	8(0.8808)	6(1.0000)
6	9(0.1554)		
7	8(0.1192)		
8	9(0.4223)		
9			

Step 4:

Table X_r Arrays

r	1	2	3	4	5	6	7	8	9
X_r	0	0	0	1	1	1	1	1	1

Table: N Array

n	4	5	9	8	6
-----	---	---	---	---	---

Step 5:

Table: Rearranged Array N

d_n^*	3	5	5	7	7	10
n	4	5	7	8	6	9

Table: Ordered Pointer Array

Node, n	Predecessor nodes, m, and corresponding $P_r(m, n)$		
4	5(0.4683)	7(1.0000)	
5	9(0.4223)	8(0.8808)	6(1.0000)
7	8(0.1192)		
8	9(0.4223)		
6	9(0.1554)		
9			

Steps 6 & 7:

Table: Node Weighting Array and Calculation of $V_{41}(h-1)$

n/h	4	5	7	8	6	9
W_n (initial)	1.0000	0	0	0	0	0
Updated W_n		0.4683	1.0000	0.4125	0.4683	0.1978
(= W_n)				0.5317		0.4223
						0.4951
T_{h-d}				2000	4000	1000
$V_{ij}(h-1) = W_h T_{h-d}$				1063.4	1873.2	495.1

Table: Output

Load on Link (4,1)	Origin Node	Destination Node
1063.4	8	1
1873.2	6	1
495.1	9	1
Total: 3431.7		

3. 12 The Greater Manchester Transportation Unit (GMTU)

The GMTU model, Randle (1979), incorporates a deterministic method of producing multiple routing through a network. Calculations of exact probabilities of link usage are facilitated by restricting to two, the number of links via which paths from a particular origin may arrive at a network node. Consider the movements from origin node h to nodes d_1 and d_2 in the network Figure 1.3.1 below.

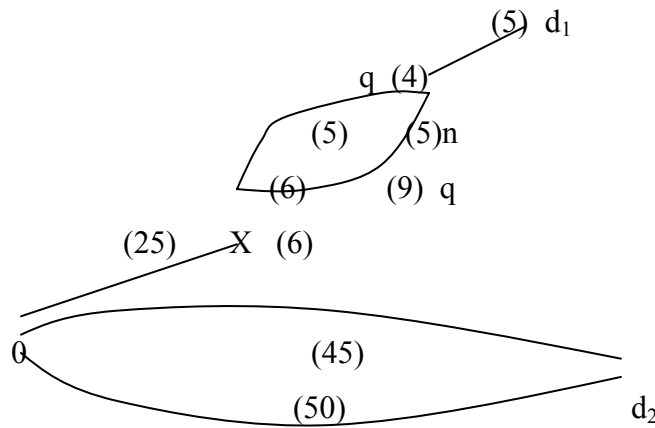


Figure 1.3.1: A simplified Network for illustration of GMTU Model

Note: Figures in brackets are costs on sections.

For both movements, there is a choice between two routes with costs of 45 and 50 units respectively. However, in travelling from h to d_1 the choice of route is actually made at node x and is based on costs of 15 and 20 units, which is likely to lead to a split between the two routes quite different from that arising from a choice between routes with costs of 45 and 50 units. It is clearly desirable to that in any route choice model, the choice between routes should be based on mutually exclusive components, an aim which Dial's nor Gunnarsson's model attempt to achieve. To accommodate this and other requirements the GMTU model is based on two considerations:

1. travelers do not all perceive identical routes and hence there should be a wide range of sensible routes between every O-D pair wherever possible; and
2. it should be possible to calculate the probability of usage of every link in the sensible paths and such calculations should consider only mutually exclusive sections of paths used.

To satisfy the first consideration, it is assumed that on any link with actual cost \bar{c} , the distribution of perceived costs is uniform with mean \bar{c} and range $2k\sqrt{\bar{c}}$, where the spread factor, k , is chosen by the model user.

If, in accordance with the second consideration, choices are to be based on mutually exclusive sections of route, then they should be between pairs of routes; choices between pairs of routes also facilitates direct calculation of route usage probabilities.

In the GMTU model, this is accomplished by storing, for each node n having two or more predecessor nodes, both the minimum cost path predecessor node, p , herein referred to as the 'best' predecessor node, and the second-best predecessor node, q . The minimum paths from the origin node, h , to p and q are then retraced from p and q towards h until they meet, at node x say, as in figure 1.3.1.

Apart from the fact that at most two predecessor nodes are stored for any node, the only other restriction on route usage is one which is necessary for the vine loading algorithm; if the minimum path cost to q is greater than the minimum path cost to n , then the probability of usage of link (q,n) is set to zero. This measure, which is effectively the same as Dial's requirement that on all links used, the initial node must be nearer to the origin than the final node, also prevents U-turns loops.

GMTU Method of Assignment Using the Data from a Tree Table with Two Predecessor Nodes

In the following, it is assumed that a Tree Table from origin h is available, containing, for each node n:

- i. The best predecessor node, p, and the minimum path cost from h to n via node p, denoted by $c_{n(p)}^*$;
- ii. The second-best predecessor node q, c_q^* (the minimum path cost from h to q) and $c_{n(q)}^*$ (defined as for node p above), if there is no second-best predecessor node then q, c_q^* and $c_{n(q)}^*$ are all zero; and
- iii. If required, R, the order of removal of node n from the loose ends table.

Step 1: (Determination of the probability of usage of link (p,n))

For each node n in the Tree Table: -

- a. From the read values of p, $c_{n(p)}^*$, q, $c_{n(q)}^*$ and c_q^* ;
- b. If q=0 or $c_q^* > c_{n(p)}^*$ ($= c_n^*$) set $P_r(p,n) = 1.0$; otherwise:
 - i. Using the minimum path predecessor nodes in the Tree table, retrace the minimum cost paths to p and q backwards, towards h, until they meet, at node x say, and read c_x^* from the table;
 - ii. Calculate the 'actual' cost from x to n via p and q by

$$\bar{c}(p) = c_{n(p)}^* - c_x^* \text{ and } \bar{c}(q) = c_{n(q)}^* - c_x^* \text{ respectively; and,}$$

- iii. if $\bar{c}(p) + k/(\bar{c}(p)) < \bar{c}(q) - k/(\bar{c}(q))$, set $P_r(p,n) = 1.0$, else set

$$P_r(p,n) = 1.0 - \frac{[\bar{c}(p) - \bar{c}(q) + k\{1/(\bar{c}(p)) + 1/(\bar{c}(q))\}]^2}{8k^2 / (\bar{c}(p) \cdot \bar{c}(q))}$$
 if $q \neq 0$ and $1.0 - P_r(p,n) < \text{suitable small value}$, set q=0 in the tree table and set $P_r(p,n) = 1.0$.

Step 2: (Initialization of the nodal volumes V_i and link volumes V_{ij})

- a. i. set $V_i = 0$ for all nodes i ; and
 - ii. set $V_d = T_{h-d}$, the number of trips from h to destination node d , for all destination nodes d .
- b. For all links (i,j) , set the assigned link load $V_{ij} = 0$.

Step 3: (Assignment of the trips from origin h to the network).

Starting with the node, n , having the greatest value for the order of removal, R , i.e. with the greatest value of c_n^* , and proceeding in order of decreasing R until node h is reached, for each node n :-

- a. Read p and q from the Tree table;
- b. Set $V_{pn} = V_n P_r(p, n)$ and set $V_p = V_p + V_{pn}$
- c. If $q \neq 0$, set $V_{qn} = V_n - V_{pn}$ and set $V_q = V_q + V_{qn}$.

Note: - If the secondary analysis may be required at some future date, the values of $P_r(p, n)$ should be added to the Tree Table.

Example 3:

Using the GMTU assignment model with $k=1.2$, assign 100, 300, 500 and 600 trips from origin node to destinations 3,4,5,and 7 respectively in the network shown in the figure 1.3.2 below. Assume that the ‘actual’ costs, shown in () on the links are the same in both directions.

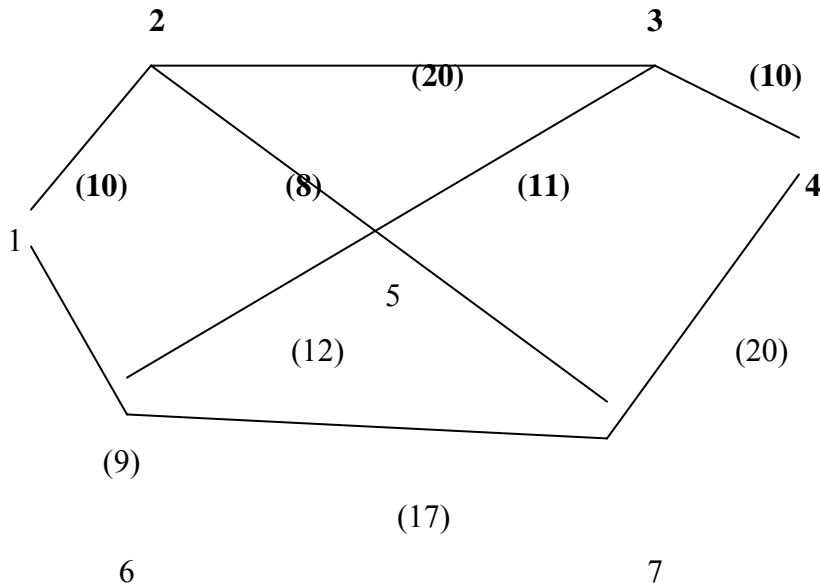


Figure 1.3.2 A simple Network for the GMTU Assignment Example 5

Table 1.3.1 Network Table

Link No a	Initial Node	Final Node	Link cost, c_a	Link No a	Initial Node	Final Node	Link cost, c_a
1	1	2	10	11	5	2	8
2	1	6	9	12	5	3	11
3	2	1	10	13	5	6	12
4	2	3	20	14	5	7	7
5	2	5	8	15	6	1	9
6	3	2	20	16	6	5	12
7	3	4	10	17	6	7	17
8	3	5	11	18	7	4	20
9	4	3	10	19	7	5	7
10	4	7	20	20	7	6	17

Table :Tree Table

Node n	Best Node, p	Predecessor $c_{n(p)}^* (= c_n^*)$	Second- best Node, q	Predecessor $c_{n(q)}^*$	F Array	Order of Removal, R
1	0	0		0	1	1
2	1	10	0	999	3	3
3	5	29	2	30	6	6
4	3	39	7	45	9	7
5	2	18	6	21	11	4
6	1	9	0	999	15	2
7	5	25	6	26	18	5

Step 1: Calculation of $P_r(p, n)$

Table :Calculation of $P_r(p, n)$

Node n	Predecessor Nodes		Intersection Node, x	c_x^*	$\bar{c}(p)$	$\bar{c}(q)$	$P_r(p, n)$
	p	q					
	1	0	0				
2	1	0					1.0
3	5	2	2	10	19	20	0.5898
4	3	7	5	18	21	27	0.8801
5	2	6	1	0	18	21	0.7428
6	1	0					1.0
7	5	6	1	0	25	26	0.5791

Step 3: Link Loading

Table :Node Loading (a)

Node No. n	Node Load, V
1	1500.1(b)
2	953.8
3	364
4	300
5	1083.0
6	267.8
7	636.0

Note:

- (a) Nodes in the order of decreasing R: 4, 3,7,2,6,1
- (b) Difference from 1500 due to rounding

Table: Link Loading

Initial Node, i	Final Node, j	Link Load, V_{ij}	Initial Node, i	Final Node, j	Link Load, V_{ij}
1	2	953.8	5	2	0
1	6	546.3	5	3	214.7
2	1	0	5	6	0
2	3	149.3	5	7	368.3
2	5	804.5	6	1	0
3	2	0	6	5	278.5
3	4	264.0	6	7	264.0
3	5	0	7	4	36.0
4	3	0	7	5	0
4	7	0	7	6	0

Selected Link Analysis for the GMTU model

To check whether or not trip from origin node h contribute to the flow on any link (i,j), it is necessary to check whether i is a predecessor node to j in the Tree Table constructed when determining minimum paths from h. For this, Randle and Turner (1979) can be used. Basically, the Tree Table for the origin node h is transformed into a forward pointing list of sub-lists, the sub-list for each node, n, being either empty (when n has no successor nodes i.e. n is not a predecessor node) or consisting of its successor nodes (i.e. those nodes which do have n as one of their predecessor nodes).

The procedure for determining the movements from the origin node h loading link (i,j) is:

Step 1:

From the Tree table, check that i is a predecessor node to j ; if it is, store $P_r(i, j)$, otherwise, consider the next origin node or stop.

Step 2: (Setting up the pointer array)

- c. Initialize the pointer array. In each row enter 0 for the first node and 0.0 for the corresponding probability.
- d. For each node (row), n , in the Tree Table: -
 - i. Read p, q and $P_r(p, n)$ and enter n and $S(n, p) = P_r(p, n)$ in row of p of the pointer array, (if $p = \text{zero}$, i.e. n is the origin node, consider the next node in the tree table);
 - ii. If $q \neq 0$, enter in row q of the pointer array, node n and $S(n, q) = 1 - P_r(p, n)$

Step 3: (Setting up array N consisting of nodes, n , which may be arrived at on paths from h passing through node j)

Step 4: (Setting up an 'ordered' pointer array)

Arrange the nodes n in N order of increasing c_n^* and enter n together with the entries in row n of the pointer array in the 'ordered' pointer array.

Step 5: (Setting up a node weighting array, w , containing for each node n in N , the probability that a path to n from h passes along link (i, j)).

- a. Initialize array W .
Set $W_j = P_r(i, j)$; set $W_n = 0$ for all other nodes in N .
- b. From each row in turn of the 'ordered' pointer array: -
 - i. read the 'pointing' node n ($n=j$ initially), and the list of pairs of successor nodes s and probabilities $S(s, n)$; and

- ii. for each of the nodes ‘pointed’ to, calculate $W_s = W_n \cdot S(s, n)$ and set

$$W_s = w_s + W_s$$

Step 6: (For all destination nodes, d, calculation of the trips to d from h passing along link (i,j) and out put of the link loading information).

- a. From the trip matrix, read off the destination nodes, d, and T_{h-d} , the trips from h to d.

- b. Consider each destination node, d, in turn: -

If d is not in array N , consider the next destination node, if any, else, if d is in N :

- i. calculate $V_{ij}(h-d) = W_d \cdot T_{h-d}$; and
 ii. out put a message indicating that there are $V_{ij}(h-d)$ trips on link (i,j) from the origin node h to destination node d.

Example 4:

In Example 5, determine the movements from node 1 loading link (1-6)

Step 1: From the Tree table 1.3.2, in the row for the node $j=6$, $p=1=i$. From table 1.3.3, row $j=6$, $P_r(1,6) = 1.0$. (Note that $P_r(i, j)$ values would normally be stored in the Tree table if any secondary analysis is anticipated).

Step 2:

Table: Pointer Array

Node (row), n	Successor nodes, s, and corresponding (s(n)) τ
1	2(1.0000) 6(1.0000)
2	3(0.4102) 5(0.7428)
3	4(0.8801)
4	0.0
5	3(0.5898) 7(0.5791)
6	5(0.2572) 7(0.4209)
7	4(0.1199)

τ Values of $S(s, n) = P_r(n, s)$ are taken from table 3.3.

Step 3:

Table 1.3.7: x Array

r	1	2	3	4	5	6	7
x_r	0	0	1	1	1	1	1

Table 1. 3.8: N Array

n	6	5	7	3	4
----------	---	---	---	---	---

Step 4:

Table: 'Ordered' N Array

c_n^*	9	18	25	29	39
n	6	5	7	3	4

Table: 'Ordered' Pointer Array

Node, n	Successor nodes, s & (S(s,n))	
6	5 (0.2572)	7 (0.4209)
5	3 (0.5898)	7 (0.5791)
7	4 (0.1199)	
3	4 (0.8801)	
4	0 (0.0000)	

Step 5:

Table: Node weighting Array & Calculation of $V_{16}(1-d)$

n/d	6	5	7	3	4
Initial W_n	1	0	0	0	0
Updated W_n		0.2572	0.4209	0.1517	0.0683
(=W_s)			0.5698		0.2018
T_{h-d}		500	600	100	300
$W_d T_{h-d}$		128.6	341.9	15.2	60.5

Table: Output

Load on Link (1,6)	Origin Node	Destination Node
128.6	1	5
341.9	1	7
15.2	1	3
60.5	1	4

Comments on the Dial, Gunnarsson and GMTU Models

- GMTU model is similar to Dial's in that all links included in paths have their initial node 'closer' than their final node to the origin node h but, in that the probability of usage of any link is determined at the node at which the choice of including the link arises.
- In GMTU model, the division of the trips from origin node h arriving at any node i between its predecessor links (i, j) etc is based on explicitly on the costs on mutually exclusive sections of the minimum cost paths from h to i via these predecessors; the probability of usage being zero for links other than those with initial node k as the best or second-best predecessor node to i . Gunnarsson splits the trips leaving any node i for destination node between the successor links (i, j) on the basis of the costs on the minimum paths from i to the destination node d via these links.
- In using the multinomial logit model as the basis for the division of trips between routes, Dial's model implicitly assumes constant variance in perceived costs on all route sections; the same is true of Gunnarsson's model if the exponential form is used for $f(r_{ij})$.
- Route choice probabilities in Dial's method are based on numbers of routes available, ignoring completely the fact that many routes may be identical apart from minor diversions. Gunnarsson's and GMTU make choices of links to be used at the nodes at which the choice arises and are also sensitive to the network structure.

3.13 SCATA, an Arithmetical Approximation Approach to Multiple Route Choice

If there are relatively few route choices, all of which can be explicitly enumerated, arithmetical approximation methods of estimating route choice probabilities, and hence link flows, are more accurate than simulation methods. Such circumstances are extremely scarce in traffic assignment studies and hence the techniques are rarely used in practice; however, they can be used to provide an insight into the characteristics of solutions obtained by the simulation techniques that are not apparent from their application.

SCATA assumes a uniform distribution. Robertson and Kennedy (1979) suggested that the assignment of traffic in a network by simulation methods is relatively insensitive to the shape of the probability distribution assumed. SCATA, first proposed by Robertson (1977) and then refined by Robertson and Kennedy (1979), has a composite cost structure

Simple Choice Algorithm for Trip Assignment (SCATA)

In the two route choice situation with uniform distributions of perceived link costs, the proportions of trips choosing the dearer route, R_2 , given in terms of the 'actual' route costs \bar{c}_1 , \bar{c}_2 and spread factor, k , may be written in terms of \bar{c}_1 and \bar{c}_2 and the ranges of the perceived cost distributions, r_1 and r_2 , as:

$$P_r(R_2) = (\bar{c}_1 - \bar{c}_2 + (r_1 + r_2)/2)^2 / (2r_1r_2) \text{-----}(11)$$

Assuming that the perceived costs on competing routes are independent of one another, the variance in the perceived cost on R_1 relative to R_2 , called the relative variance, is:

$$Var_{1,2} = Var_1 + Var_2 \text{-----}(12)$$

where Var_1 and Var_2 are variances of perceived costs on R_1 and R_2 respectively.

As approximation, it may be assumed that this relative variance is divided equally between the two routes, so that the variance of the perceived cost on each link is:

$$Var = 0.5(Var_1 + Var_2) \text{-----(13)}$$

Assuming that the common range of the distribution is r then, since $Var = r^2 / 12$,

$$r^2 = 6(Var_1 + Var_2) = 6Var_{1,2} \text{-----(14)}$$

Substituting r from the above equation for ?

$$P_r(R_2) \cong 0.5(1 - (\bar{c}_2 - \bar{c}_1) / \sqrt{(6Var_{1,2})})^2 \text{-----(15)}$$

Note: $P_r(R_2) = 0$ if the term in parenthesis is negative. This facilitates the elimination of unused routes.

Choice between three or more parallel routes

The calculation can be simplified by converting the single problem in n dimensions to $(n-1)$ problems in two dimensions. This is achieved by a procedure which enables a pair of parallel links to be replaced by an equivalent single link. Values are calculated for the ‘the reduced’ cost’ and ‘reduced variance’, the mean and variance respectively of the distribution of perceived costs on the equivalent link, so that the traffic assignment in the remainder of the network is unaffected by the substitution.

Reduced Cost

Assuming $\bar{c}_1 - 0.5r_1 \leq \bar{c}_2 - 0.5r_2 \leq \bar{c}_1 + 0.5r_1 \leq \bar{c}_2 + 0.5r_2$, the probability density function for the distribution of perceived costs, c , on links (routes) actually used is:

$$f(c) = 1/r_1 \text{ for } \bar{c}_1 - 0.5r_1 \leq c \leq \bar{c}_2 - 0.5r_2 \text{-----(16)}$$

and $f(c) = (\bar{c}_1 + \bar{c}_2 + 0.5(r_1 + r_2) - 2c) / (r_1 r_2)$ for $\bar{c}_2 - 0.5r_2 \leq c \leq \bar{c}_1 + 0.5r_1$

The mean perceived ‘cost’ on the route actually chosen, i.e, the ‘reduced cost’, is given approximately by:

$$\bar{c}_{1+2} \cong \bar{c}_1 - \sqrt{\{4(P_r(R_2))^3 Var_{1,2} / 3\}} \text{-----(17)}$$

This indicates the important fact that the mean perceived cost of travel on routes actually used through a network is less than the mean perceived cost of travel on the minimum cost path.

Reduced Variance

Robertson and Kennedy (1979) gave the following approximation on the ‘reduced variance’:

$$Var_{1+2} \cong (1 - 1.3P_r(R_2))Var_1 + P_r(R_2)P_r(R_2) + 0.2Var_2 \text{-----(18)}$$

From the above equation, if $P_r(R_2) = 0.5, Var_{1+2} = 0.35(Var_1 + Var_2)$

In general, the variance of perceived costs on routes actually used is also less than the variance of perceived costs on the minimum path.

Example 1 :

1000 vehicles per hour travel from h to d between which the three possible routes R_1, R_2 and R_3 , with no overlapping sections. The distributions of perceived costs on the routes are assumed to be uniform with means, \bar{c} , of 1.0, 1.2, and 1.4 equiv. Mins on R_1, R_2 and R_3 respectively, and ranges $2k\sqrt{\bar{c}}$ where $k = 1.2$ when \bar{c} is measured in tenths of equiv. Mins. Estimate the mean and variance of the ‘reduced’ costs on the routes actually used and the flow of vehicles on each route.

In the following, costs are in units of tenths of equiv.mins.

Since $Var(c) = (range)^2 / 12 = (2k\sqrt{\bar{c}})^2 / 12 = 0.48\bar{c}$, we have:

$$\bar{c}_1 = 10; Var_1 = 4.8; \bar{c}_2 = 12; Var_2 = 5.76; \bar{c}_3 = 14; Var_3 = 6.72.$$

Consider the replacement of routes R_1 and R_2 by the equivalent route e ,

$$Var_{1,2} = 4.8 + 5.76 = 10.56$$

$$P_r(R_2) = 0.5\{1 - (12 - 10) / \sqrt{(6(10.56))}\}^2 = 0.2803$$

$$\bar{c}_e = \bar{c}_{1+2} = 10 - \sqrt{\{4(0.2803)^3 (10.56) / 3\}} = 9.443$$

$$Var_e = Var_{1+2} = (1 - 1.3(0.2803))(4.8) + 0.2803(0.2803 + 0.2)(5.76) = 3.826$$

Consider the replacement of e and 3.

$$Var_{e,3} = 3.826 + 6.72 = 10.546$$

$$P_r(R_3) = 0.5\{1 - (14 - 9.443) / \sqrt{(6(10.546))}\}^2 = 0.0912$$

$$\bar{c}_{e+3} = (1 - 1.3(0.0912))(3.826) + 0.0912(0.0912 + 0.2) 6.72 = 3.551$$

Thus the mean perceived cost on the routes actually used is $\bar{c}_{e+3} = 0.934$ equiv.mins.

The variance of the 'reduced' costs on the routes actually used is,

$$\bar{c}_{e+3} = 3.551 \times 10^{-2} (\text{equiv. min } s)^2$$

The flow on R_3 is $1000 \times 0.0912 = 91.2$,

The flow on R_2 is $10000(1 - 0.0912)(0.2803) = 254.7$,

The flow on R_1 is $1000(1 - 0.0912)(1 - 0.2803) = 654.1$ veh/h.

Table: The Effect of the Order of Combing Routes when Traffic is Assigned to three routes in parallel (Example 7)

Proportion of Traffic Routes R_1, R_2 and R_3	Order of combing Routes			Max. Deviation from Average (%)
	(1+2)+3	(1+3)+2	(2+3)+1	
$P_r(R_1)$	0.6541	0.6429	0.6635	1.6
$P_r(R_2)$	0.2547	0.2571	0.2370	5.0
$P_r(R_3)$	0.0912	0.1000	0.0995	9.4
Reduced Cost	9.340	9.330	9.312	0.2

In Langdon (1982), it is suggested that if it cannot be assumed that the introduction of an additional route (mode in fact) does not change the relative proportions of trips on the other routes, then it is necessary to carry out the calculations three times with each route in its turn becoming the last one introduced.